

# Development of a system for real time localization of a team of athletes using Ultra Wide band.

Design choices and considerations for mesh network localization. UT supervisors: Prof. Dr. P.J.M. Havinga, Dr.

D.V. Le Viet Duc, Ir. E. Molenkamp  
Company supervisor: Ir. R. van Os

J. Gerth

May 13, 2019

## Abstract

This research covers the entire development of an Ultra Wide Band (UWB) based localization system. We start with the distance estimation technique and do experiments to determine system parameters. Since the system is a mesh network, a way structuring communication is developed which includes synchronization of devices in the field. Suitability of different radios for this synchronization is examined using experiments. This results in a mesh of devices generating device to device distances which are broadcast using Bluetooth Low Energy (BLE). These distances are fed into a localization algorithm using Multidimensional Scaling (MDS) to estimate a location of devices.

The result of this research is a working, real world mesh localization system using UWB Time Of Flight (TOF) measurements. We find that accuracy is mainly influenced by the number of nodes in the system and the ratio of  $range/fieldsize$ .

A working mesh localization system is the result. Performance is mainly influenced by the number of nodes in the system and the ratio of  $range/fieldsize$ . The system does not achieve the required accuracy on a large field with few devices in the field due to this latter limitation.

These findings are consistent with previous research where separate parts of a system were evaluated. These researches did however not study these effects when using a fully working system, but rather simulated such a system using only 2 devices and performing point to point distance measurements. This research shows the effect of several variables on a real mesh network and shows the limitations of such a network.

**Index terms**— Ultra Wide Band (UWB), Mesh Network, Localization, Relative distance estimation, Design, Athletes, Tracking

## Introduction

Tracking athletes can be done using GPS when the playing field is outdoors. Indoors GPS does not work and some other technology has to be used. UWB is a viable candidate as its characteristics make it possible to precisely timestamp messages [1]. Research has already been

done on using UWB for localization and some commercial products are already on the market [2–12]. Focussing on the sports solutions, the existing products all use devices which are fixed to a wall or ceiling. This allows the devices to be wired to a central control station and precisely synchronized. Our goal is to provide a flexible system which can be deployed anywhere to track athletes in real time. Having said that, the system is not limited to athletes but can also be used during disasters to track emergency personnel for example.

The proposed system consists of anchors which have a known location, for instance on the corners of the playing field. And blind nodes, or tags, for which we want to estimate the location. The system is accurate enough for tracking athletes when the ratio of  $range/fieldsize$  is not too small and when there are enough blind nodes in the field. This means that the field used must not be too large, or the devices must have a long reliable UWB communication range. Localization accuracy can be greatly improved by using previously estimated locations. This is however not fully covered in this paper as the experiments were done with a non moving setup. Using historical data would cause the localization solution to quickly converge to the actual position thereby making the results of different experiments too similar to compare.

Further limitations of the system are scalability. Due to the use of BLE advertising to communicate results, no more than 10 devices with an update rate of 1 Hz can be used at this point. It is expected that this can be solved by using another communication method but this has not been tried.

## Thesis Outline

First in section 2 we will present the work that has already been done on the different aspects of this research. After that we will introduce the used hardware and a few limitations of it in section 3. Then in section 4 we will explain different distance estimation techniques and accompanying messaging schemes. At the end of that section the system is able to estimate distances between devices. To have a structured way of determining which mesh device at what point in time can estimate distances, a protocol is developed and presented in section 5. The protocol also

describes the contents of messages. Synchronization of the mesh devices is presented in section 5.3. The localization algorithm which estimates a location from the measured distances is explained in section 6 and in section 7 the system is put to the test. Conclusion and recommendations can be found in section 8 and section 9 respectively.

## Research question

Preliminary research is done in [13] and from this the research question has arisen.

How to design a suitable architecture and protocol for an UWB real time localization system to achieve optimum performance in terms of accuracy and responsiveness, while using minimal resources and achieving scalability?

# 1 Requirements

To answer the research question, some requirements for the system are needed. In the rest of this section these requirements are discussed and quantified.

## 1.1 Accuracy

Accuracy is defined as the inverse of the error a localization step has for a reference device. So a lower error means a higher accuracy. Since the implementation is to track a team of athletes on a playing field, the needed accuracy is linked to the dimensions of an athlete. We do not want to see if the athlete is just outside the lines or for instance offside. The main goal is to track the amount of distance covered and position in the field of the athlete. Since an athlete is about one metre this seems a good upper limit for the error. This is the error of the location estimation from distance measurements alone. Predictions based on previous and measured speed and acceleration can further improve accuracy.

## 1.2 Responsiveness

Responsiveness is the update rate of the location estimation. Also this can be specified for the application on athletes. This update rate also puts an upper bound on the latency of  $1/\text{update rate}$ . For the application of tracking athletes, firstly the system could be used to prevent injuries due to fatigue. Secondly the system could be used to display entertaining statistics about each athlete such as distance travelled, maximum speed, maximum acceleration and more. The next location of an athlete, or anything for that matter, can be predicted quite accurately if there is no acceleration. An athlete in a team sport will not have significant velocity changes in a second therefore an update rate of  $1\text{ Hz}$  is required. Note that this is the update rate of the locations estimated from distance estimations alone, additional measurements such as acceleration can be used to predict between updates.

## 1.3 Scalability

Scalability is essentially the number of athletes that can be tracked. If we take a football team of 11 players, add one or two referees and place 6 anchors along the field we already need almost 20 devices. This is a good start for the minimum supported number of devices which might later be increased. If the system is scalable we can increase the number of players using the same design.

# 2 Related work

UWB is not a new technology and is already commercially available. It is used widely for tracking people or objects [9–11, 3]. There are companies which sell a system for tracking athletes using UWB ranging [14, 12]. However these systems are mounted to the walls/ceiling of the room they are deployed in and thus are not very flexible. In [15, 16] a survey of recent research in UWB indoor positioning is given.

In 2002 a prototype localization system is developed to assess the capability to perform in severe multipath environments [7]. This paper mainly focusses on the effect of obstacles and reflective surfaces in the measurement area. The system consists of wired anchors and mobile tags. The latter transmits packages asynchronously, resulting in collisions and an update rate of  $1/5\text{ Hz}$ . In [17] a prototype for a localization system using UWB is presented. This however uses only three anchors and one tag. The tag sequentially polls the anchors for distance measurements and distances are passed from the tag to a PC using UART. The localization algorithm is only executed when a distance to all anchors is measured in the correct order. This results in a non scalable and definitely not wearable system.

In [18] a localization system for indoors environments is proposed. This paper mainly focusses on the digital implementation of a transmitter and receiver and an estimation algorithm for localization. A robot using UWB localization for navigation is realized indoors. This research presents a system where a single mobile tag is localized by only using distances to anchors.

Mesh network localization is also studied extensively. The research mostly focusses on cheap sensors being deployed in a remote area where a very crude localization is acceptable. A simple Ad-Hoc positioning system is evaluated in [19]. This, and many others, uses (a version of) DV Hop localization which extracts the range from the number of hops a data packet must make. This technique is popular since it does not require specialized hardware but only uses a communication infrastructure [20]. In [21] the performance of relative ranging using UWB technology in 1D and 2D is evaluated using static tags. This research focusses on the effect of the number of hops a tag is away from an anchor on the accuracy of location estimation.

In [22] the theoretical bound on accuracy of localization in a wireless sensor network is studied. The researchers derive Cramér–Rao bounds for different scenario’s and compare this with a real world experiment. They find that location estimation variance bounds decrease as more devices are added to the network. In [23] two UWB devices

are used to measure point to point distances. A device is placed at the first point and a second device at an other point. Range measurements are performed and the devices are moved to different locations. With this a distance matrix is created and a maximum likelihood estimator is used to determine a location.

Sharing a common communications channel requires a protocol. Implementations range from saying “over” when a person is done using a radio channel to coding information with a bit pattern so it can be resolved on the receiving end. One of the first “protocols” researched is the ALOHA protocol, described in [24]. This protocol is based on acknowledgement of reception and re-transmitting if no acknowledgement is received.

The MDS algorithm is already used for localization in sensor networks. In [25] a new algorithm is proposed in which measurements that are believed to be more accurate are weighted more, contributing more to the solution. In [26] a nonmetric MDS algorithm is proposed to improve accuracy over the classical MDS algorithm.

As presented above, the components of the system developed in this research are not new. UWB is already used for distance measurements and MDS is known for its capabilities in localization. It is however that all these researches only cover a small part of the development of a system presented in this research. In [23], which is the closest thing to the system presented in this research, no actual mesh network is used. The researchers only prove that UWB devices can produce a matrix of measurements which can be solved for locations. It does not however take into account the constraints of a real world system, such as time necessary for distance estimation, synchronization requirements and communication errors. This research provides a system to show how UWB localization performs in real world scenarios.

## 3 Hardware

Even though this research is not focussed on hardware, it is an important part of the system. When doing research on UWB transceiver IC’s one keeps showing up: the DW1000 from Decawave.

This IC is already used for commercial and industrial Real Time Location System (RTLS) solutions [10, 27, 28]. This widespread use means that it probably performs well and there is a lot of information to be found. First some features of the IC will be highlighted in section 3.1. This, and the fact that the IC’s and modules containing the DW1000 are easily obtainable via online stores made the choice for the DW1000 an easy one. A preliminary study has been done using a DWM1000, a module containing the DW1000 IC. Results will be discussed in section 4.2. Information will be given on the hardware used for the actual experiments in this paper, the MDEK1001, in section 3.2.

Devices are numbered 0-11, and determine their own ID by looking up the PART ID [29] in a lookup table.

### 3.1 DW1000 IC

The DW1000 IC features everything that is needed for UWB communication on a cheap IC. This chip is controlled via SPI and thus usable with a standard low cost microcontroller. The chip itself however is quite extensive and requires a lot of settings to be correct for it to work. Features of this model are [30, 29]:

- Range  $\approx 300\text{ m}$ .
- Precision of  $\approx 10\text{ cm}$ .
- Wireless synchronization between anchors.
- Programmable transmit output power.
- SLEEP and DEEPSLEEP mode with low power consumption.
- Supports Time Difference of Arrival (TDOA) and Two-Way ranging.
- Channel diagnostics.

To explore the capabilities of the DW1000 two modules and microcontroller development boards were bought. The modules were easily connected to the microcontroller’s SPI bus using wires. This offered a great start in seeing what the chip can do and how it works. The results of this experiment are discussed in section 4.2.

#### 3.1.1 Time stamp capabilities

UWB has great timing potential due to the short pulses being used [2]. The DW1000 uses this potential to precisely timestamp transmitted and received messages. The timestamps are 40-bit values at a nominal 64 GHz resolution, for approximately 15 ps event timing precision [30]. In 15 ps light, and thus the radio waves used, travels around 0.5 cm. In the DW1000 IC the calibrated antenna delays can be programmed to increase accuracy of timestamping.

#### 3.1.2 Crystal Drift

The crystal for the DW1000 on the DWM1001 module, which is at the heart of the MDEK1001, is trimmed in the factory to  $\pm 3\text{ PPM}$  [31]. This means that in one hour the time recorded by a crystal can drift  $(60 \cdot 60 \cdot 10^3) \cdot (3/10^6) = 10.8\text{ ms}$ . In this time light, and thus a ranging packet, travels about 3 km.

The crystal for the nRF52832 high speed system clock has an accuracy of  $\pm 10\text{ PPM}$  according to the BOM in [32]. This means that in one hour the time recorded by a crystal can drift  $(60 \cdot 60 \cdot 10^3) \cdot (10/10^6) = 36\text{ ms}$ .

### 3.2 MDEK1001

While the DWM1000 modules connected to a microcontroller development board were a good start to get familiar with the chip, they were not really robust and easy to use. Luckily Decawave offers another UWB development kit, containing 12 MDEK1001 devices. This features an nRF52832 microcontroller, the DW1001 IC, antenna, accelerometer, power supply, IO and programming interface all in one package. This development kit offers another

feature next to the ease of use: BLE. However this device only has a range of 60 *m* according to Decawave [33]. Since in section 4.2 it was found that the reliable range is lower than this no problems are expected.

## 4 Distance estimation

Distance estimation, also called ranging, lies at the core of the system. It are these estimated distances that eventually are used to calculate (relative) positions. Distances can be determined by measuring different quantities. Exactly what is measured to estimate a distance is what we call the technique from now on. Within each technique messages have to be exchanged to send information from one device to another. The number of messages and what they should contain is called the scheme. The technique is discussed in section 4.1 and the scheme in section 4.4.

### 4.1 Ranging Technique

Ranging techniques can be divided in two main categories: those using a Time Of Flight (TOF) measurement and those without. We will discuss a non-TOF technique for completeness and give a reason to discard this. After that we will discuss the TOF technique.

#### 4.1.1 RSS

In Received Signal Strength (RSS) distance estimation a message is sent from device A and received by device B which measures the RSS. If the pathloss model is known, a distance can be estimated. The RSS estimation in the DWM1001 however is not very accurate as explained in [29], especially for higher levels of RSS and thus shorter distances. This inaccuracy and the fact that Ultra Wide Band (UWB) is favoured due to it's timing capabilities non-TOF based techniques are discarded.

#### 4.1.2 TDOA

TOF based techniques include Angle Of Arrival (AOA), Time Difference of Arrival (TDOA) and Time Of Arrival (TOA). These techniques rely on precisely measuring time between reception and transmission of messages or frames. The DW1000 chip is capable of this precision which makes it ideal for use in TOF techniques.

The TDOA technique works, as the name implies, on the difference in time of reception of a frame. A tag broadcasts a message which is received by multiple anchors. These anchors are synchronized in time and can calculate the difference in time between reception on devices. The difference in time is directly related to the difference in distance which results in a parabola between the two anchors on which the tag must be. A visual description can be seen in Figure 4.1a, here the green filled parabola represents the line on which tag B must lie, based on the difference in arrival of a message in anchors P1 and P2. The major advantage is that the amount of messages needed is only one per distance estimation to all the anchors in range.

This however requires the anchors to be precisely synchronized in time and keep this synchronization. Due to

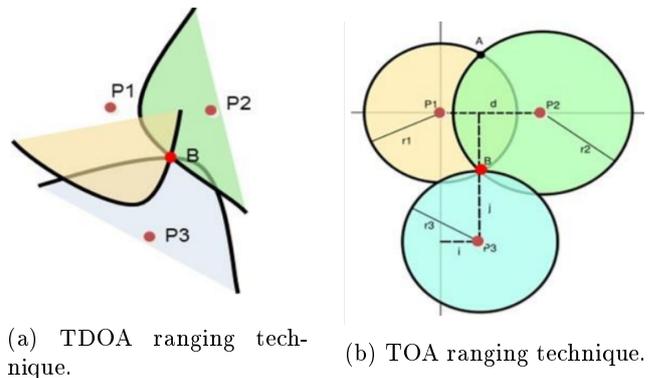


Figure 4.1: Ranging techniques based on TOF [34].

the clock drift explained in section 3.1.2 this synchronization is not kept. This means that the clocks of the devices have to be synchronized often. This can be done if the anchors are wired to a central point but this becomes harder if the anchors are stand alone since it has to be done wireless. Decawave does provide a paid solution for this, which is not used in this research.

#### 4.1.3 AOA

AOA can for instance be done as a local TDOA with multiple receivers in one anchor. Receivers measure the difference in arrival of a frame and together with the distance between these receivers an angle can be calculated. Synchronization in this case is easy since the receivers can be wired to the same PCB. Another implementation of AOA can be done using multiple directional antennas of which the RSS for each antenna can be measured. From these RSS values a direction of the message can be determined.

#### 4.1.4 TOA

Another technique which does not rely on a time difference of arrival is the TOA technique. Sent and received messages are timestamped and from these timestamps a TOF is calculated. Sending a message containing the send time of that message is preferable, else a second message has to be sent containing this information. The DW1000 has a delayed send feature, this means that the transmit time can be calculated beforehand and stored in the message. The hardware then makes sure that the message is sent at the time calculated. Using this delayed send feature makes it possible to send a message with it's own transmission time as payload.

This TOF corresponds to a distance of the responder to the initiator. With the distance to three known points a location can be determined as shown in Figure 4.1b. Here P1, P2 and P3 are the static anchors, and  $r_1$ ,  $r_2$  and  $r_3$  is the distance of a device to each anchor respectively. This technique has the advantage that the clocks of both devices do not need to be synchronized since every device measures it's own time. The disadvantage however is that multiple messages are needed for each range estimate which negatively affects scalability.

## 4.2 Preliminary results

To explore the capabilities of the DW1000 IC, two DWM1000 modules were bought. These modules include everything that is necessary for UWB communication. The modules were connected to a microcontroller SPI port using wires. Decawave provided software examples, which were ported to the used microcontroller for testing. After the software was ported, the first distance measurements could be done.

Although the results were quite accurate with a mean error of less than 25 *cm* and no increase in mean error with distance. The standard deviation did increase significantly with distance. The most interesting result however is the limited range of the DWM1000. With the settings that should result in the longest range, 40 *m* was the maximum distance for reliable distance estimation. With the settings that provided the fastest data transfer, 20 *m* was the maximum distance between the two devices for reliable distance estimation. Reliable means that the estimation was successful, and did not fail due to lost messages

This short range means that a simple setup where tags only estimate a distance to anchors on the side of the field does not work. Simply because not the entire playing field can be covered from only the side. This led to the development of the distance estimation relative to other devices in the field.

## 4.3 Choosing the technique to implement

The RSS technique is a waste of the resources of the DW1000 chip since it does not use the precise timing circuitry which is one of the main features of the chip. Besides that, the RSS method is very inaccurate as explained in [29]. The measurements done in the preliminary research in section 4.2 have shown that the standard deviation of the RSS value is quite large (more than 1 *dB* at 25 *m*).

If a receiving device can measure the AOA but the orientation of the device is not known, the direction of that message can not be related to a line in the field. Thus AOA only works when the orientation of the receiving device is known. Since the system will use a mesh network this is not an easy to use technique. It has no advantages over other techniques, therefore it will not be used. It can however be used in the stationary anchors along the field to improve the location accuracy.

Therefore the choice is between the TDOA and TOA method which both use the precise timing capabilities of the DW1000 chip. As said before, the TDOA technique requires synchronization. This can be done via wires, which is impossible if a mesh network is desired since all players have to be connected to the wired network. The wireless solution from Decawave is not used, so this solution is also not implemented.

This means that TOA is the only technique remaining, which is the one to be implemented. One of the disadvantages is that multiple messages are necessary for estimating a single range. How much messages and the consequences of this will be discussed in section 4.4.

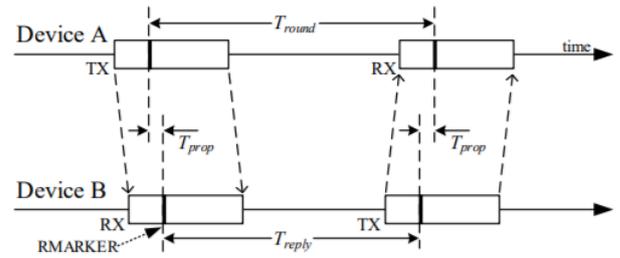


Figure 4.2: Graphical representation of SS-TWR [29].

## 4.4 Ranging Scheme

With the suitable technique chosen in section 4.1 a message scheme is to be selected. This means that we have to choose how a distance estimation is performed using the Time Of Arrival (TOA) technique. When choosing a ranging scheme, characteristics that matter are:

- Number of messages in scheme: If a range estimation in a certain scheme requires a lot of messages it might be useless as the responsiveness or scalability is impacted.
- Accuracy: Some schemes are more accurate than others i.e. due to their sensitivity to clock drift.

In this chapter, first a few ranging schemes are presented and discussed. In the end a choice is made.

### 4.4.1 Single Sided Two Way Ranging

A Single Sided Two Way Ranging (SS-TWR) scheme is shown graphically in Figure 4.2. Device A sends a message and stores the transmit time of that message. When device B receives that message, it stores the receive time. Device B then calculates when the response can be sent and puts both receive timestamp and transmit timestamp in a response message. It then sends the message on the pre-calculated time which is received and again timestamped by device A.

From the two timestamps in device B the reply time  $T_{reply}$  is calculated, and from the two timestamps in device A the total round trip time. The reply time is subtracted from the round trip time which results in the total time the messages have been in the air, so twice the Time Of Flight (TOF).

This ranging scheme only uses two messages for each distance estimation. It is however not very accurate if  $T_{reply}$  in Figure 4.2 becomes longer or the clock errors increase. The effect is examined in section 4.4.2.

To make the SS-TWR scheme scalable an adaptation is made to decrease the number of messages even further. In this adaptation one device instantiates the ranging message exchange by sending an initiation message. Devices that receive this message respond after a predetermined delay. This delay is different for each device so the responses do not overlap and the initiating device can receive all responses correctly. This means that the last device will have the longest delay and thus probably the lowest accuracy, as shown in section 4.4.2.

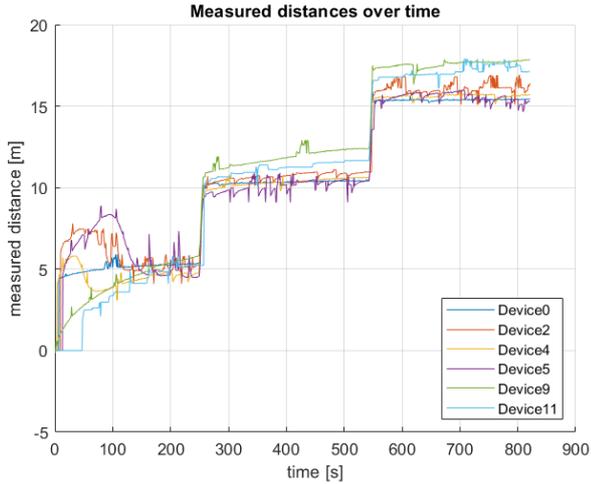


Figure 4.3: Measured distances using the SS-TWR scheme at 5 – 15 m. Failed measurements are filled with previous value, outliers are rejected.

#### 4.4.2 Performance degradation of Single Sided Two Way Ranging with longer reply times

To test the impact of  $T_{reply}$  on distance estimation an experiment has been performed. All devices are given an index which is linked to a Universally Unique IDentifier (UUID) of the DW1000 IC. The  $T_{reply}$  of each device is set according to  $T_{reply} = 1500 + ID \cdot 1000 \mu s$ , giving each device a unique  $T_{reply}$ . The devices only respond to messages from device 1, which initiates a distance measurement each second. After calculation device 1 sends the distances via Bluetooth Low Energy (BLE) to a PC. The devices were placed on a table, 5 m apart and after 5 minutes were moved to 10 m and again after 5 minutes to 15 m without stopping the experiment.

The results of the distance measurements are shown in Figure 4.3. It can be seen that the measurements at 5 m have a large deviation from the actual distance. This is because the chip is warming up and there is no compensation for this change in temperature. At the distances of 10 and 15 m, when the device has already warmed up, measurements are much more steady and some interesting observations can be made. First and foremost it can be seen that device 0 performs the best in terms of estimating a distance and, although not shown, has by far the lowest standard deviation of all devices. It can also be seen that in general devices with a higher ID, which have a longer  $T_{reply}$ , perform worse than devices with a lower ID and thus shorter  $T_{reply}$ . This performance degradation with longer  $T_{reply}$ , is due to the crystal drift explained in section 3.1.2. In this case in the devices with a longer reply time the crystals have a longer time to drift which negatively impacts the accuracy of a distance measurement.

The effect of the performance degradation is device specific. This means that device 11, which has a longer  $T_{reply}$  than device 9, does not necessarily perform worse than device 9. This is because the crystal error is *somewhere* in the  $\pm 3$  PPM range and thus the crystal of device 11 could be better calibrated and thus drift less during  $T_{reply}$ . This can also be seen in Figure 4.3 where device 11 performs

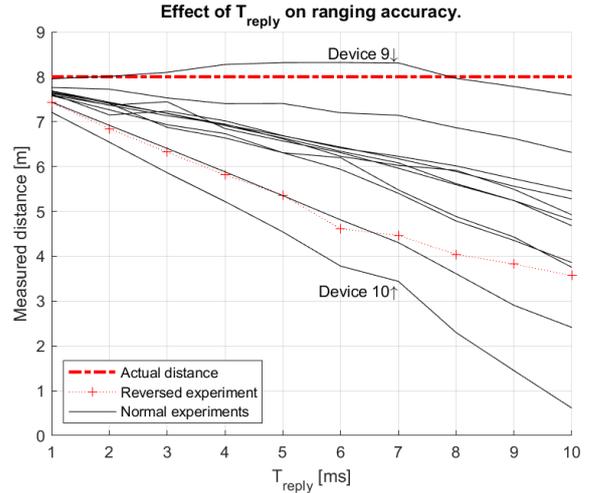


Figure 4.4: The effect of  $T_{reply}$  on distance measurements. Displayed is the mean of the measurements, per device. Also displayed is the reversed experiment.

better than device 9.

To gain a better insight in the effect of  $T_{reply}$  another experiment has been performed. This time the devices are permitted to warm up for a period of 20 minutes and then a one to one distance estimation is performed. These are always initiated by device 0, at a rate of approximately 8.5 Hz. Results are collected for a minute and then the answering device (device B in Figure 4.2) is reprogrammed for another  $T_{reply}$  and collection is started again. A table with the statistical data for each measurement can be seen in Table A.1.

In Figure 4.4 the results for each device are plotted. It can be seen that although the severity varies from device to device the length of  $T_{reply}$  has a significant impact on accuracy of the distance measurements. We can also see that when  $T_{reply}$  is longer than 2 ms the accuracy of 1 m is, on most devices, not achievable.

What stands out in Figure 4.4 is the fact that all, with exception of device 9, underestimate the distance. It is expected that this is due to device 0 having the slowest crystal. An experiment has been performed to validate this. The experiment is similar to the one performed before, but it is “reversed”. Device 10 initiates the ranging and device 0 answers. The results can be seen in Figure 4.4 as ‘...+...’. This measurement was flipped around the 8 m line and shows that the expectation of device 0 having the slowest crystal was correct.

The ‘...+...’ line in Figure 4.4 does not lie exactly on the line of device 10 from the normal experiment. This is due to the crystals not having the same absolute crystal error and the times that are measured on both devices are different in length. In the “normal” experiment device 0 measures  $T_{round}$  which is longer than  $T_{reply}$  and thus the crystal on device drifts more. In the “reversed” experiment, device 10 is used to measure  $T_{round}$  and because this crystal has a smaller error it drifts less.

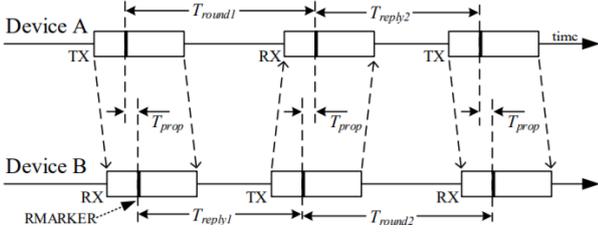


Figure 4.5: Graphical representation of DS-TWR [29].

#### 4.4.3 Double Sided Two Way Ranging

Double Sided Two Way Ranging (DS-TWR) is essentially two SS-TWR range estimations, first initiated by device A and after that by device B. In this scenario on both devices a  $T_{round}$  and a  $T_{reply}$  is measured which increases accuracy. This scheme uses 4 messages for a distance estimation. Decawave has proposed a variation on DS-TWR which uses only three messages [35]. A graphical representation of this scheme is given in Figure 4.5.

The calculations for the TOF are more complicated than for the SS-TWR or DS-TWR with four messages case. The calculations are taken from [35] and are repeated here for completeness.

For reasons of brevity not all equations are given if for instance  $R_a$  is given it is up to the reader to derive  $R_b$ . To relate the following equations to Figure 4.5 the equalities in Equation 4.1 should be used.

Equation 4.2 is an ideal representation of the round trip time, if we introduce a clock error ratio  $k_a$  such that  $\hat{R}_a = k_a R_a$  and similarly  $\hat{D}_a = k_a D_a$  we get an estimate of the durations, as a function of the real durations and a clock error. If we implement this in Equation 4.2 we get Equation 4.4 since  $D_b = \hat{D}_b/k_b$ .

Multiplying Equation 4.4 by  $\hat{R}_b$ , writing out the product and do some cleaning up gives Equation 4.6.

Subtracting  $\hat{D}_a \hat{D}_b$  and dividing by  $\hat{R}_a + \hat{D}_a$  but on the right side using the written out version  $2T_f k_a + \frac{\hat{D}_b k_a}{k_b} + \hat{D}_a$  gives Equation 4.7. On the right side, take out  $2T_f k_b$  and multiply by  $k_b/k_b$  gives Equation 4.8 where the fraction on the right hand side is equal to 1. Therefore it can be written as Equation 4.9a.  $k_b$  is expected to be very close to one<sup>1</sup> so Equation 4.9a is an accurate estimator for the TOF.

$$\begin{aligned} R_a &= T_{round1} \quad \text{and} \quad D_b = T_{reply1} \\ R_b &= T_{round2} \quad \text{and} \quad D_a = T_{reply2} \end{aligned} \quad (4.1)$$

$$R_a = 2T_f + D_b \quad (4.2)$$

$$\hat{R}_a = k_a R_a = k_a (2T_f + D_b) \quad (4.3)$$

$$\hat{R}_a = 2T_f k_a + \frac{\hat{D}_b k_a}{k_b} \quad (4.4)$$

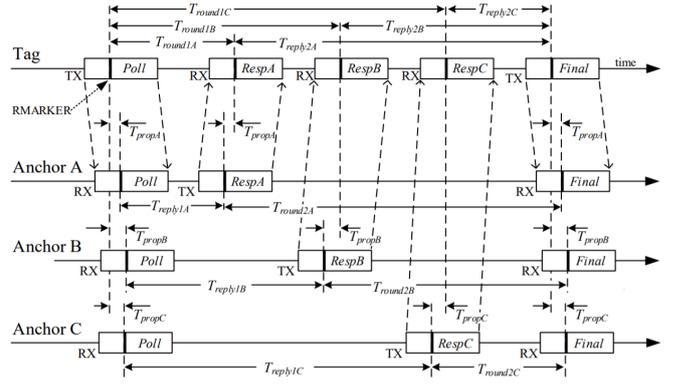


Figure 4.6: Graphical representation of DS-TWR with multiple responders [29].

$$\hat{R}_a \hat{R}_b = \left( 2T_f k_a + \frac{\hat{D}_b k_a}{k_b} \right) \left( 2T_f k_b + \frac{\hat{D}_a k_b}{k_a} \right) \quad (4.5)$$

$$\hat{R}_a \hat{R}_b = 4T_f^2 k_a k_b + \hat{D}_a \hat{D}_b + 2T_f (\hat{D}_a k_b + \hat{D}_b k_a) \quad (4.6)$$

$$\frac{\hat{R}_a \hat{R}_b - \hat{D}_a \hat{D}_b}{\hat{R}_a + \hat{D}_a} = \frac{4T_f^2 k_a k_b + 2T_f (\hat{D}_a k_b + \hat{D}_b k_a)}{2T_f k_a + \frac{\hat{D}_b k_a}{k_b} + \hat{D}_a} \quad (4.7)$$

$$\frac{\hat{R}_a \hat{R}_b - \hat{D}_a \hat{D}_b}{\hat{R}_a + \hat{D}_a} = 2T_f k_b \frac{2T_f k_a k_b + \hat{D}_a k_b + \hat{D}_b k_a}{2T_f k_a k_b + \hat{D}_b k_a + \hat{D}_a k_b} \quad (4.8)$$

$$\frac{1}{2} \frac{\hat{R}_a \hat{R}_b - \hat{D}_a \hat{D}_b}{\hat{R}_a + \hat{D}_a} = T_f k_b \approx T_f \quad (4.9a)$$

$$\frac{1}{2} \frac{\hat{R}_a \hat{R}_b - \hat{D}_a \hat{D}_b}{\hat{R}_b + \hat{D}_b} = T_f k_a \approx T_f \quad (4.9b)$$

In Equation 4.9a and Equation 4.9b (which is obtained by swapping a's and b's) all the clock error ratio's have cancelled out except for one, which makes this implementation very robust to clock errors. Equation 4.9a is only dependant on the clock error ratio of device B, while Equation 4.9b is only dependant on the clock error ratio of device A. This means if device B has a more precise clock, Equation 4.9a should be used to estimate the TOF.

However if both devices are the same, the average should be used since the TOF estimation is as good as, or better than the worst estimation. Decawave has experimentally found an approximation to calculate the average of TOF calculations from both devices which is given in Equation 4.10 which is also from [35].

$$T_f = \frac{\hat{R}_a \hat{R}_b - \hat{D}_a \hat{D}_b}{\hat{R}_a + \hat{R}_b + \hat{D}_a + \hat{D}_b} \quad (4.10)$$

DS-TWR can also be adapted for a situation with multiple responders, an example is shown in Figure 4.6. This results in more accurate distance estimation compared to the situation presented in section 4.4.1 as specified in [36].

<sup>1</sup>In a crystal with 10 PPM accuracy  $0.99999 < k_b < 1.00001$

## 4.5 Choosing the scheme to implement

The three considerations for this scheme are accuracy, responsiveness and scalability. To find an optimum for the implementation some boundaries have to be set for some requirements where others can be optimized. In section 1.1 and section 1.2 it is reasoned that a minimum accuracy of 1  $m$  and an update rate of 1  $Hz$  should be achieved.

The scheme presented in the last part of section 4.4.1 is the most scalable since it adds the least messages for each device. It does however suffer from accuracy degradation as explained in section 4.4.2. From Figure 4.4 we can conclude that  $T_{reply}$  should be less than 2  $ms$  to achieve this accuracy, assuming that the devices that are used for testing are a representative set for all DW1000 devices. Now we have an upper limit for  $T_{reply}$  we must know how many devices can be reached within this period.

When using the configuration of the “reversed” experiment in section 4.4.2 the  $T_{reply}$  is reduced to the lowest value with which the distance estimation still performs reliable. This number is 750  $\mu s$  in this configuration. The main limitation in reducing the constant in  $T_{reply}$  is the HPDWARN error. This error is raised if the time at which the DW1000 needs to send the message lies more than half a clock period (around 8.5  $s$ ) in the future and thus the DW1000 needs to wait for the clock to wrap around. Now that the minimal  $T_{reply}$  for the first answer is estimated,  $T_{reply}$  for the second and sequential messages has to be estimated. Multiple devices are now used in which  $T_{reply} = 750 + ID * x$  where  $x$  is to be estimated. Again an experiment has been performed. It is found that with  $x$  set as 1000  $\mu s$ , 750  $\mu s$  and 500  $\mu s$  the amount of range estimations that succeeded are 96.0 %, 93.4 % and 91.1 % respectively. With a smaller  $x$ , for instance 400  $\mu s$  the success rate drops to 50 % which is unusable. We fix  $x$  at 600  $\mu s$  to have a little safety margin. This means that three devices can respond to a single initiating message, since the longest  $T_{reply}$  is now  $750 + 2 * 600 = 1950$ . This is just under the 2  $ms$  limit set before and distance estimation to the latest device that answers should be accurate enough. Devices that answer to an initiation message are put in groups of three, so that each device knows if and after what time to answer an initiation message.

With the settings determined above the SS-TWR should be accurate within 1  $m$ . It also requires only four messages for a distance estimation to three devices. The DS-TWR with multiple responders shown in Figure 4.6 does only require one message more but makes the error due to  $T_{reply}$  possibly smaller as explained in [36]. However since implementing this does not fit in the planning it is discarded as an option and SS-TWR is used. It will however be treated in section 9.

## 5 Protocol

In the mesh network we need a way to have communication between the devices without collision of packets. We need a channel access method as the channel has to be accessed by multiple devices. The channel access methods are named after the different resource they divide amongst

the devices. These methods include Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), Code Division Multiple Access (CDMA), Space Division Multiple Access (SDMA) and Power Division Multiple Access (PDMA) [37].

In our mesh network we need to reach as much devices as possible in order to have better chances of estimating enough distances to a device such that a location can be solved. This means that devices can not use other frequencies or only send messages to certain areas of the playing field by using a directed antenna. The one thing that can actually be divided over devices is time. This means that the TDMA method is the method of choice.

Time is divided in superframes which are further divided into timeslots. A superframe is the smallest block of actions that is repeated. In this case it contains a distance estimation between all devices and communicating the results. Each device gets its own timeslot in which it performs distance estimation to neighbours and communicates results. As we decided in section 1.2 that an update rate of 1  $Hz$  is enough, the superframe length is set to 1 second. This superframe is divided into 20 timeslots according to section 1.3 which leaves us with a timeslot of 50  $ms$  for each device.

In this timeslot a device estimates the distances to its neighbours, using the method described in section 4. These estimated distances are communicated to the pc running the localization algorithm using BLE as described in section 5.2. After a device’s timeslot is done it listens for other devices performing distance estimations.

TDMA relies on all devices having a common timebase. Therefore we need all devices to use the same clock or have all their clocks synchronized. If Global Positioning System (GPS) would be available that time could be used as common clock. As this is not the case, synchronization is implemented and described in section 5.3.

### 5.1 Ranging Messages

The mesh network communicate between themselves using Ultra Wide Band (UWB) which are used for both distance estimation and synchronization. As described in section 4.5 ranging requires an initialization and response message. The message contents are described here.

#### 5.1.1 Initialization message

This is the message sent by the device whose timeslot it is. It contains some info about the message, which group it is meant for and also the system time of the sender. Bytes are ordered as in Table 5.1. Receiving devices especially check the **For group** field, match this with their own assigned group number and respond only they match. This is to divide the devices in groups to ensure the short response times as specified in section 4.5.

System time is sent in  $\mu s$  so with 32 bits the maximum value represents more than 70 minutes. If this value overflows there is no problem as only the least significant bits matter. We only need the devices to be synchronized on millisecond level to determine how far along the superframe the mesh network is. When this is known a device

Byte	0	1	2	3	4	5
Content	Message info	For group	System time bits 7-0	System time bits 15-8	System time bits 23-16	System time bits 31-24

Table 5.1: Contents and ordering of an initialization message used for ranging.

Bit	0	1	2	3	4	5	6	7
Content	Is init message	Is syncmaster message	Not used					

Table 5.2: Contents and ordering of message info byte.

can either initiate distance estimations in it’s own timeslot or listen for incoming messages.

The message info contains information about the message and is further specified in Table 5.2. The **Is init message** field is used so that devices only take action when an initialization message is received as devices also receive response messages from other devices. Bits 2-7 are not used yet and are reserved for future use.

### 5.1.2 Response message

When an initialization message meant for the device is received a device responds with a response message. This message is structured as specified in Table 5.3.  $T_{reply}$  is calculated and sent in the response message.  $T_{reply}$  is in DWT time units, this means if we only send 32 bits the maximum value that can be represented is  $2^{32} / (128 \cdot 499.2 \cdot 10^6) = 0.067 s$ . As the maximum value for  $T_{reply}$  is 2 ms we should not need more than 32 bits.

Byte	0	1	2	3	4	5
Content	Message info	Device ID	$T_{reply}$ bits 7-0	$T_{reply}$ bits 15-8	$T_{reply}$ bits 23-16	$T_{reply}$ bits 31-24

Table 5.3: Contents and ordering of an response message used for ranging.

Byte	0	1	2	3	4	5	...	30	31
Content	GAP data length	GAP data type	Sequence number bits 7-0	Sequence number bits 15-8	Integer part distance to device 0	Decimal part distance to device 0	...	Integer part distance to device 13	Decimal part distance to device 13

Table 5.4: BLE advertising packet payload contents and ordering.

## 5.2 Data communication

We need a way to communicate the estimated distances to the localization algorithm. The UWB radio is not suitable a direct communication since the range is not sufficient. Therefore we use the other radio which has a longer range, namely the BLE radio.

Since we only need to send small amounts of data it is not required to set up a bluetooth connection. Instead we use advertising packets to send data from the mesh to the PC running a localization algorithm. Mesh devices advertise their packets which contain distances at the fastest possible rate. This rate however is once every 20 ms which is the minimum interval as specified in [38]. With the current timeslot length this means that the packet is advertised at most 2 times.

Packet payload structure is shown in Table 5.4. The first two bytes are defined by the Bluetooth Special Interest Group [39] and specify the length and type of data (30 bytes and manufacturer specific data respectively). After this, since we use manufacturer specific data type (0xFF) as the data type we would normally put a two byte company identifier (0x0059 for Nordic Semiconductors). This identifier is byte swapped so byte 2 would be 0x59 and byte 3 would be 0x00. Byte 2 is filled with a sequence number identifying the message, ranging from 1 to 255.

Measured distances are placed in the message with each distance using 2 bytes. The integer part has a range of 0-255 and the decimal part of 0-99. The distance is to a specific device, which is indicated by the place in the advertising message. This means that at this point only distances to device 0-13 can be advertised. See section 9 for an explanation on how to send more distances.

These advertised packets are received by an nRF PCA10040 development board which filters on BLE addresses and only acts if the received message is from a mesh device. Another filter is deployed which keeps track of the last 10 received device ID and sequence number combinations. If the combination is already in the last 10 messages the received message is not forwarded. This is to prevent from forwarding the same message multiple times. Before forwarding, the byte at position 0 is replaced by the

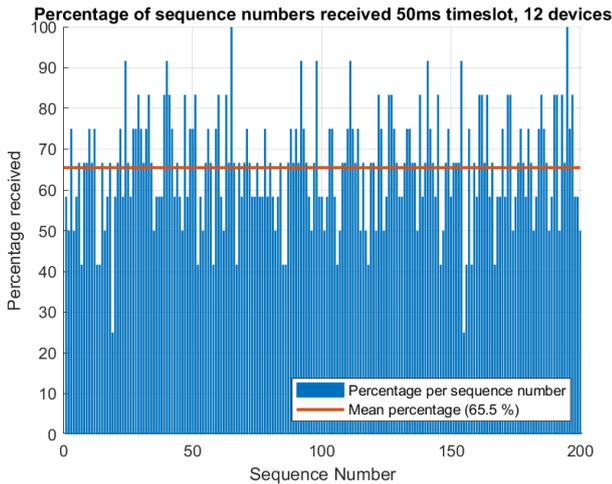


Figure 5.1: Performance of BLE communication with a timeslot length of 50 *ms*.

device ID of the sender deleting GAP data length.

### 5.2.1 Performance of data communication

To assess the reliability of communication as implemented, an experiment has been performed. Devices behave as normal: sending out a measurement packet via BLE at the end of their timeslot. Each timeslot the sequence number present in the messages is incremented. Messages are recorded for 200 seconds at a distance of 1 *m* and the sequence numbers are counted. If sequence number *x* is counted *y* times we have successfully received measured distances from *y* devices. The result of this is scaled to the number of devices used in the experiment and shown in Figure 5.1.

It can be seen that the percentage of received BLE packets is quite low. At 65.5% this means that we lose a third of the distances measured<sup>2</sup>. This is a substantial amount and probably has a significant negative effect on performance of the localization algorithm.

To solve this problem we have increased the timeslot length. We first changed it to  $1000/12 \approx 80$  *ms* which is the longest where we can still accommodate 12 devices. After this the timeslot length was changed to 100 *ms* which means that only 10 devices fit in a timeslot. The experiments have been performed the same as before. For comparison an experiment with 10 devices and a timeslot of 50 *ms* has also been included. The results of this experiment can be seen in Table 5.5 where it is clear that a longer timeslot has a positive effect on the number of received BLE messages. This is expected as the advertising messages are now sent multiple times, increasing the chance of reception.

Elongation of the timeslots does have the desired effect in that the reliability of BLE communication improves a lot. It does however come with an added disadvantage in that it affects the scalability. With a timeslot length of 100 *ms* the success rate of the communication is acceptable but the amount of devices is limited to 10. A possible

<sup>2</sup>If all devices measure the same amount of distances, which we assume for now

Length timeslot	Number of devices	% received
50 <i>ms</i>	12	65.5%
50 <i>ms</i>	10	63.0%
80 <i>ms</i>	12	75%
100 <i>ms</i>	10	87.8%

Table 5.5: BLE communication performance.

solution to increase communication performance and keep scalability is given in section 9.

## 5.3 Synchronization

As explained in section 3.1.2 the clock of the nRF52832 can drift by as much as 36 *ms* in an hour. This is quite a lot since each device only gets 50 *ms* of time to perform distance estimation and communicate results as explained in section 5. If no action is taken to prevent this clock drift, or to mitigate the effects, devices could send simultaneously resulting in colliding packets. These collisions can be prevented by building in time buffers between frames where no device is allowed to send. The clocks could then drift, resulting in the devices sending in those buffer times. But if these buffers are long enough such that no collisions should occur in the worst case of two clocks drifting in opposite direction these buffers have to be at least  $2 * 36$  *ms* long to have the system running for an hour. This would eat away a very significant proportion of the available time and is therefore not a realistic solution.

A way to synchronize the systems clocks on the nRF52832 is needed. Since the UWB radio already present on the device has a high time resolution, this would be a perfect candidate. However when the synchronization signal is coming from a single point in the playing field and with low range of the UWB it is expected that not all devices would receive the signal.

Another radio already present on the MDEK1001 is a BLE radio, which should have more range than the UWB radio. This radio can advertise its own system time on which receiving devices can sync their clocks. For this to work the time between fetching current time from the system clock on the sending device and a receive event on the receiving device has to be constant and thus predictable. An experiment to find these timing characteristics has been performed and can be found in section 5.3.1. For the UWB radio the same experiment has been performed in section 5.3.2.

From these experiments it is clear that BLE advertising is not usable for synchronization and the timing characteristics of the UWB packets indicate usability for synchronization. However the limited range of UWB excludes the use of a single static “master device” alongside the field periodically sending out synchronization messages. A solution for the synchronization algorithm is presented in section 5.4.

### 5.3.1 Reception timing of BLE advertising packets

This experiment uses three MDEK1001 as receiving devices and one PCA10040 nRF development board as advertising device. The setup is quite simple: on the advertising device a pin is set high when the time is retrieved from the system clock. This time is put in a message and advertising is started. The advertisement is configured that it sends a single BLE advertising packet. On the receiving end, scanning for BLE packets is continuously enabled. When a packet is received from the sending device an interrupt is triggered which also sets a pin high.

Advertising and scanning is abstracted in a SoftDevice provided by Nordic Semiconductors. Parameters are configured and advertising/scanning is enabled with a call to this SoftDevice. It is this SoftDevice which triggers the interrupt.

The pins are connected to an oscilloscope which is connected to a laptop. On this, a Python script waits for the oscilloscope to be triggered on the sending pin going high and then checks if the receiving pins are also asserted. If all pins have a rising edge, which means all devices have received the transmitted message, the data is logged. The oscilloscope samples at  $100\text{ kS/s}$  and thus has a resolution of  $0.01\text{ ms}$ .

100 measurements are performed and the results can be seen in Table 5.6. It can be seen that the time between sending and receiving (indicated by the pin becoming high) varies enormously. Since the timing is handled in the SoftDevice we do not have any influence on the exact timing of these packets.

Device number	1	2	3
Maximum	13.91 ms	14.08 ms	13.95 ms
Minimum	2.67 ms	2.55 ms	2.55 ms
Range	11.24 ms	11.53 ms	11.40 ms
Mean	8.66 ms	8.46 ms	8.64 ms
Standard deviation	3.08 ms	3.01 ms	3.06 ms

Table 5.6: Statistical data of measured delay between retrieving system clock and generating an interrupt on receive for BLE advertisements.

What is not shown in this table is that the difference in receiving time between multiple devices varies much less. In fact in the measurements performed this difference was never more than  $2\text{ ms}$ . If all devices would always receive all the advertisement packets this would be precise enough. However more often than not devices miss a packet. This means that if device A corrects its time on a packet after a minimal delay and device B corrects it on a packet after maximum delay in Table 5.6 the two device clocks would differ by more than  $10\text{ ms}$ . This amount of clock difference is not usable and therefore synchronization via BLE advertising packets is abandoned.

### 5.3.2 Reception timing of UWB packets

BLE is not a viable candidate and wired or GPS synchronization not an option. Another option is synchronization via UWB. To find the timing characteristics of sending

and receiving UWB packets, the same experiment as in section 5.3.1 is performed but now with the UWB radio. The sample frequency of the oscilloscope is set to  $100\text{ MS/s}$  resulting in a resolution of  $0.01\text{ }\mu\text{s}$ . Bluetooth radio is disabled completely since this would also be disabled in the actual implementation at the time of distance estimation. Results of this experiment can be seen in Table 5.7

Device number	1	2	3
Maximum	458.56 $\mu\text{s}$	458.73 $\mu\text{s}$	459.94 $\mu\text{s}$
Minimum	438.27 $\mu\text{s}$	437.71 $\mu\text{s}$	438.66 $\mu\text{s}$
Range	20.29 $\mu\text{s}$	21.02 $\mu\text{s}$	21.28 $\mu\text{s}$
Mean	448.27 $\mu\text{s}$	448.48 $\mu\text{s}$	449.30 $\mu\text{s}$
Standard deviation	5.78 $\mu\text{s}$	6.33 $\mu\text{s}$	6.43 $\mu\text{s}$

Table 5.7: Statistical data of measured delay between retrieving system clock and generating an interrupt on receive for UWB packets.

It is clear that the time delay is much more constant, and thus predictable, than with the BLE advertising packets in Table 5.6. The delay is predictable within a few microseconds, this gives the opportunity to possibly synchronize on sub millisecond level. This is certainly usable for a synchronized communication and distance estimation protocol. The reason for using a mesh network is that not all devices are always within communications range of a “master device” or a static anchor alongside the playing field. This still calls for a solution to have all devices synchronized using UWB packets which is more complicated than listening for a broadcast time from a central “master device”.

## 5.4 Synchronization Algorithm

One solution for the low range of the UWB can be that for instance the referee carries a “master device” since the referee is expected to cover a lot of the playing field. Not all devices would always receive the synchronization packets, but to keep the clock offset below  $1\text{ ms}$  for instance only about once every  $90\text{ s}$  a synchronization packet has to be received.

Another, more robust, solution is to piggyback synchronization information on the messages already being send. This solution is more robust as it does not require a single device to cover the entire playing field every  $90\text{ s}$ . Techniques for this implicit synchronization are discussed in [40]. Asynchronous Diffusion, first presented in [41] has similarities with our message scheme in section 4.4 and therefore seems a logical choice.

In Asynchronous Diffusion a device asks its neighbours for their time, calculated the average value of these times and sends out the calculated value out to its neighbours to update their values [40].

In our case, a device receives initiating messages from its neighbours containing their system times. Upon reception these times are compared with the system time of the receiving device and the correction is stored. Just before the ranging timeslot of a device begins, the calculated corrections are averaged and this correction is applied to the system clock. When the device is allowed to estimate

distances to its neighbours it sends with the initiating message its own corrected time, thereby transmitting his time to its neighbours.

The performance of the synchronization is evaluated in section 5.4.2.

#### 5.4.1 Startup

To prevent the corrections to become very large during setup of the system an initial synchronization is implemented. Devices other than the so called **sync master** wait after startup from a message from the **sync master**. If this message is received, the internal system clock is reset and the time that is sent by the **sync master** is adopted as the new time. After this normal operation begins. The **sync master** immediately begins normal operation, but since the device wakes up from a reset the system clock has a very low value. This waiting for the **sync master** provides an initial synchronization.

#### 5.4.2 Performance of synchronization

To assess performance of the synchronization an experiment has been performed. Devices drive a pin high on the start of their own timeslot as explained in section 5 and drive it low again at the end of this timeslot. Thus the high period of the pin indicates when a device decides that it is its timeslot. If the devices are perfectly synchronized at the moment device  $n$  drives it's pin low, device  $n + 1$  immediately drives it's pin high.

The pins of devices 0-3 are connected to an oscilloscope and device 1-3 are powered on. At this point nothing happens since the devices wait for a message from the **sync master** which is device 0. The oscilloscope is configured such that it triggers on a rising edge, the beginning of the timeslot, of device 0. The samplerate is 10  $kS/s$  giving a resolution of 0.1  $ms$ .

Powering on device 0 starts normal operation of the mesh network, in this case consisting of only 4 devices. The data from the oscilloscope is logged for 1000 seconds, containing the timing information of device 0-4. The statistical data of this experiment can be seen in Table 5.8. The positive edge indicates the start of a device's timeslot and the negative edge the end.  $\mu$  and  $\sigma$  indicate the mean and standard deviation of this measurement respectively.

Device number	0	1	2	3
Rising edge $\mu$	0.05 $ms$	58.44 $ms$	104.80 $ms$	159.31 $ms$
Falling edge $\mu$	49.80 $ms$	104.45 $ms$	152.86 $ms$	202.71 $ms$
Rising edge $\sigma$	0.03 $ms$	1.78 $ms$	1.59 $ms$	1.51 $ms$
Falling edge $\sigma$	0.49 $ms$	1.29 $ms$	1.22 $ms$	1.46 $ms$

Table 5.8: Statistical data from 1000 accuracy measurements of the synchronization. Only devices 0-3 were powered on in this test.

From this table we can see that the devices are fairly synchronized. Although these results look promising we want to know how these statistical properties develop over time as we want to know the quality of synchronization after a long time, for instance a match. Therefore the measurements are sliced into parts of 100 seconds for which

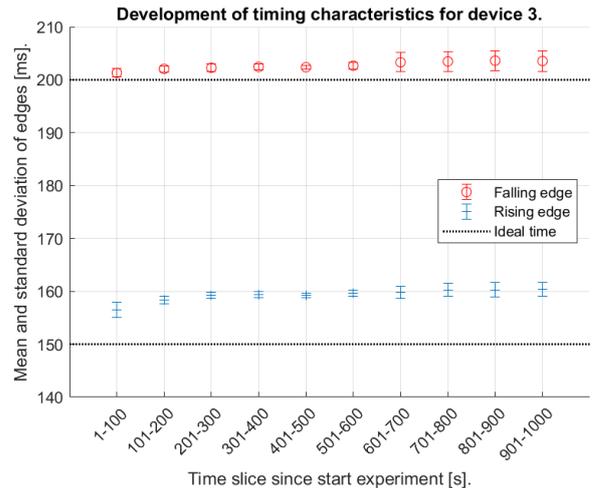


Figure 5.2: Development of the accuracy of synchronization over time.

the statistical properties are calculated. The result can be seen in Figure 5.2.

It can be seen that the initial synchronization is actually of better quality (closer to the ideal value and a lower standard deviation) than later in the experiment. But the quality seems to be stable at the end of the experiment and thus does not deteriorate. A longer experiment with all devices powered on is done and the results are in Figure B.1. There we see that although there is a deviation from ideal, it is constant. This level of synchronization is usable for this implementation although it is not ideal and should be improved.

## 6 Localization

In section 5 we have described how the data is transmitted to the localization algorithm: the BLE advertising packets are forwarded to a central PC. On this PC the advertising packets are fed into a localization algorithm which estimates positions of the device. This section is dedicated to that location estimation part of the system.

### 6.1 Gathering the data

Data is received as one line per BLE packet: data ended by the character '\n'. These lines are read by Matlab and stored in a matrix where the row corresponds to the sender of the packet and columns to the devices a distance is measured to. If the device ID of the sender is lower or the same as the device ID in the previously received packet a superframe has passed and the date is stored in a new matrix. The old matrix can then be solved.

### 6.2 Estimating locations

Finding all the pros and cons of different location estimation algorithms is a research on its own. Due to time limitations and this not being a research focussed on finding the best algorithm, we used what was available. An example which used Multidimensional Scaling (MDS) and

procrustes was provided by dr. D.V. Le Viet Duc for which we are very grateful. Since this worked sufficiently well we did not do any more research on different location estimation algorithms.

The matrices containing measured distances are transformed to a dissimilarity matrix. This means that the matrix is symmetric along its main diagonal. If two distances are measured (device A measures to B and device B measured to A) the average is taken from these two measurements. The resulting matrix is one with pairwise measured distances between devices. So, entry  $(A, B)$  in this matrix is the distance between device A and B. Either measured by one of these devices, or the average of measurements from both.

This matrix is fed into a MDS algorithm, provided by Matlab [42]. From [25]: *“The goal of multidimensional scaling is to find a low dimensional representation of a group of objects (e.g., sensor positions), such that the distances between objects fit as well as possible a given set of measured pairwise “dissimilarities” that indicate how dissimilar objects are (e.g., inter-sensor RSS)”*. We change the inter-sensor RSS for inter-sensor distance measurements.

Not all distances are always present, since some UWB or BLE packets can be lost or a device is in an error state. This means that a dissimilarity matrix can also contain NaN values. Since we use nonclassical MDS we can handle this but an initial configuration for the output has to be chosen. For the anchors we know the position, which is given as argument to the scaling algorithm. The other, unknown, positions are initially “guessed” by a random number scaled to the field size. This initial guess has a significant effect on the outcome of the MDS algorithm. In the Matlab scripts in section 7, initial guesses are always the same so results can be reproduced.

The result of the MDS algorithm has to be transformed to represent the playing field. The `procrustes` algorithm, also supplied by Matlab, is used for this transformation. This algorithm determines a linear transformation to best fit the points corresponding to the anchor locations in the output of the MDS algorithm to the real anchor locations. This transformation can then be applied to the outputs of the MDS algorithm corresponding to the blind nodes, mapping these outputs to a location in the playing field.

### 6.3 Improving localization

In a real world application, output of the previous localization could serve as an input for the initial guesses for the next iteration. This further improves the performance of the MDS algorithm to the point that in the static experiments in section 7 only a few solutions are more than 1 m away. An example of the performance increase can be seen in Figure 6.1. In this the MDS algorithm is limited to 5 iterations to show the effect of using historical data. This means that with the same number of iterations a higher accuracy can be achieved when using these previous solutions. The number of iterations in the localization algorithm is limited due to time constraints and the MDS algorithm can fail to find a fitting solution. In that case using the historical data might increase accuracy, in the

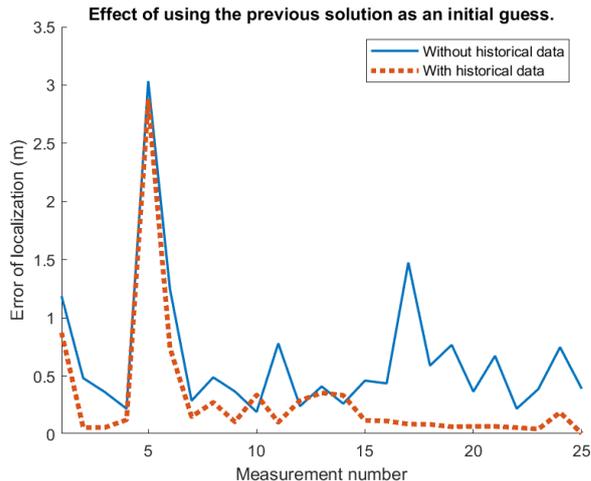


Figure 6.1: Effect of using historical data on performance of localization. The MDS algorithm is limited to 5 iterations to show the performance increase for the same number of iterations.

other cases when the iteration limit is not reached it only decreases the time needed to find a solution.

Devices in the experiments do not move and therefore the initial guess is only getting more accurate and the solution very quickly converges to one point. Historical data is not used in section 7 to show the performance of a single location estimation. Initial guesses for the current location can be improved by incorporating speed and acceleration of the previous localizations, possibly augmented by other data such as from an Inertial Measurement Unit (IMU) sensor.

## 7 Evaluation of the system

To assess the performance of the system multiple experiments have been performed. All of these experiments use pre-generated datasets. This means that the effect that is studied is isolated from other influences since it is the exact same experiment data. The generation of this dataset is explained in section 7.2. After this the experiments are explained and their results evaluated.

### 7.1 Comparison with existing work

The system developed in this research is a real world system which means that there are errors which do not occur in an idealized version of this system. The closest of a peer to peer localization system using UWB is found in [23] where two devices are used to measure the distance between multiple points. An average measured distance is determined from multiple measurements and this is fed into a localization algorithm. This means that distance estimation cannot fail, communication errors do not occur and the distance estimation error is close to zero since the standard deviation is filtered out. This means that in that research a 100% complete measurement matrix with the mean distances is the input to a localization algorithm,

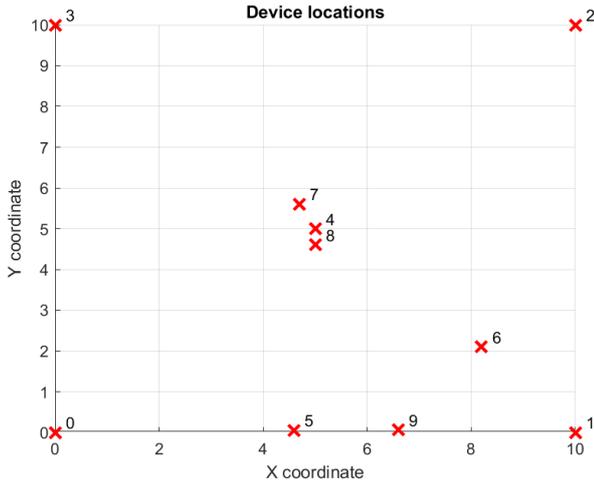


Figure 7.1: Actual positions of devices in the field.

where in our case the measurement matrix is often incomplete containing deviations from the real distances.

## 7.2 Generating dataset

For the experiment a dataset has to be generated. This is done by placing 4 devices (device 0-3) in a square of 10x10 meters. These are called the anchor nodes and their location is known to the localization algorithm. Device 4, which is the device we want to know the location of is set in the middle: at  $x = 5$  and  $y = 5$ . Device 5-9, the so called blind nodes, are placed on random locations. A random number generator is used to determine these locations. These random numbers are multiplied by the size of the field and correspond to a coordinate which can be seen in Figure 7.1. All devices are placed on a tripod at chest height. A photograph of the indoors experimental setup can be seen in Figure F.1a.

Devices are programmed to have a timeslot length of 100 *ms* to minimize the effect of unreliable data transfer using BLE as explained in section 5.2.1. Devices are turned on and allowed to warm up. After a reset, data is gathered for 500 seconds and stored. Quality of range data for the indoors experiment without a human body is shown in Table D.1.

What can be seen is that some devices clearly outperform others. Distances measured to device 0 for instance have a high standard deviation but have a mean that is very close to the real distance. Also all measurements from device 8 to 4 have failed which is probably due to the small distance between them. Devices need a minimal distance between them to perform measurements. This distance varies per device, as is shown by the fact that distance measurements measured by device 4 to device 8 are successful in 86% of the time.

### 7.2.1 Unintended benefits of using a single dataset

The dataset is manipulated to simulate real world conditions. Matrices are made smaller to simulate less devices

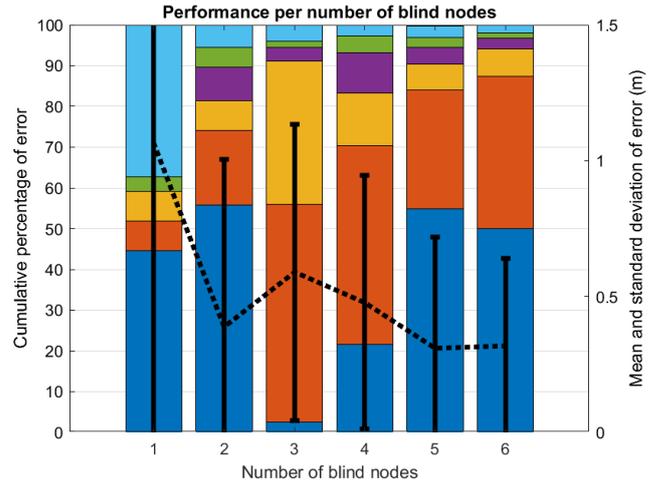


Figure 7.2: Effect of number of blind nodes on localization performance. For the legend, see Figure 7.3

and long measured distances are discarded to simulate devices that are out of range. These manipulations will be explained when they are applied for an experiment. However during generation of the dataset all devices are online and in range to one another. This means that synchronization is more precise since it receives system times from more devices than it would if not all devices are online or in range. In this section these effects are not taken into account.

## 7.3 Effect of blind nodes

The number of blind nodes that are present in the mesh are expected to have an effect on the accuracy of localization. More specific: more blind nodes mean better location estimation [43, 21, 44]. To verify this an experiment has been performed with the dataset generated in section 7.2. This datasets contains dissimilarity matrices with distances measured by all 10 devices. To simulate a matrix which would be generated by 9 devices the last row and column are deleted. This means that we always use the set of devices with the lowest device ID's that fit in the set size.

The experiment first simulates a situation where device 4 is the only blind node in the system. Then for each experiment a blind node is added until there are 6 blind nodes. The estimated location of these other blind nodes is not shown to improve comparability.

In Figure 7.2 the effect of number of blind nodes added to the system is plotted. It can be seen that the general trend is that localization accuracy increases, as expected. It is however not always the case that with more blind nodes the localization accuracy increases. For instance the accuracy with 2 blind nodes is slightly better than with 3. This is due to the initial random guess. In Figure E.1a the same experiment with the same data using an other initial guess is shown. There we can see that the overall trend is the same (error decreases with number of blind nodes) but shape of the curve is different.

This shows that the initial guess has a significant effect on performance, as explained in section 6.3.

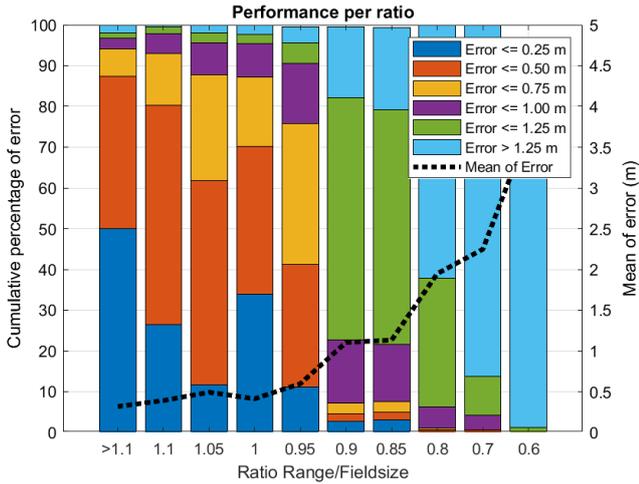


Figure 7.3: Effect of ratio of *range/fieldsize* on localization performance.

What also stands out is the large standard deviation of error of localization. This is because the measurements are seen as a single measurement. If the historical data, as explained in section 6.3 is used the standard deviation and mean error decrease significantly.

#### 7.4 Effect of simulated size field

If a field is bigger than the one used in section 7.2 not all devices are in range to one another. To see the effects a bigger field would have, distances larger than a set distance are removed from the dissimilarity matrices. This simulates the two devices where the distance is removed to not be in range. This removal is done for different ratios of range to playing field size. Since the range of these devices is somewhat consistent this corresponds to increasing the field size. This can be safely done as in [13] we have shown that quality of the measurements do not significantly degrade with distance. The main limitation in range was the UWB packets not being received any more, which is precisely what is being simulated here.

In Figure 7.3 it can be seen that the performance quickly deteriorates with ratios lower than 1. This is probably due to anchors not being able to measure a distance to other anchors. To verify this, another experiment is done. This time the measurements on the sides of the playing field, so distances between devices (0-1), (1-2), (2-3) and (3-0) are not removed. This simulates a playing field where there are also anchors on the sides of the field, and not only on the corners. Results can be seen in Figure E.1b. Mean error is the same for higher ratios because the same dataset is used. The mean error with ratios lower than 1 is lower than with the sides deleted. But more importantly: the error grows less rapidly. This might be due to the MDS algorithm having more distances to work with. But since it is a very significant increase in performance it is expected that it is because the relative distance between anchors is known.

In general, the mean error is too high when the ratio of *range/fieldsize* drops below 1. It however must be noted that in this setting the MDS algorithm is initialized only

Device	X	Y
0	0.0 m	0.0 m
1	63.7 m	0.0 m
2	63.7 m	50.0 m
3	63.7 m	99.8 m
4	0.0 m	100.3 m
5	0.0 m	50.2 m
6	31.9 m	50.0 m

Table 7.1: Coordinates of devices in the large field.

with knowledge of positions of the anchors. When the previous solution is utilized, performance should increase.

#### 7.5 Effect of size of field

For completeness, a test with a larger playing field is performed. This dataset is collected outdoors on a soccer field. Locations of the anchors and device we want to locate (device 6) can be seen in Table 7.1. Other devices are placed on the edge of the penalty area. In Table D.3 the quality of measurements between known devices is shown. What stands out is that the mean measured distance is mostly quite precise. Only device 0 has a large standard deviation, just as in Table D.1.

The more interesting thing however are the success rates, these are overall quite low. This is also what was found in [13]: above a certain distance the communication becomes less reliable. These low success rates result in a measurement matrix which is only partly filled. In section 7.8 we can see that this has an impact on the accuracy of localization. Matrices on the larger field are filled less than on the smaller field due to devices not being in the reliable communications range with each other. Localization performance is therefore significantly worse than with a more filled matrix on a smaller field.

In Figure 7.4 the effects of adding more nodes to a larger field is shown. As in the smaller field, adding more nodes increases localization accuracy. Adding more nodes however can decrease accuracy if that node does not add enough distances to the measurement matrix. The MDS algorithm has to solve a location extra and therefore needs more distances.

Due to the constraint of 10 devices by the timeslot length as explained in section 7.2 we cannot evaluate the performance when the same field is occupied by more devices.

#### 7.6 Effect of a human body

This effect can not be simulated with the dataset from section 7.2. Therefore the experiment is repeated but we put a human body in contact with device 4. At chest height where the device would also be worn in real life. The person is at  $x = 5$  and  $y \approx 5.2$  in Figure 7.1. A photograph of the experiment outdoors with a person is shown in aFigure F.1b. The rest of the experiment is not changed and thus the data from this experiment can be compared with the data obtained in section 7.2. A

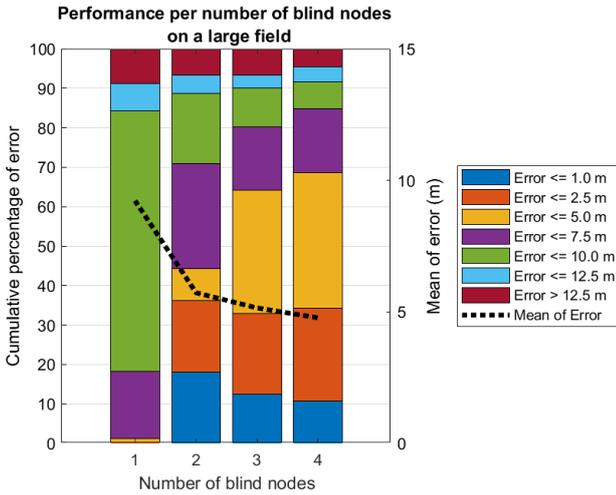


Figure 7.4: Effect of number of blind nodes on localization performance in a larger field.

comparison of an experiment with and without a human body present in the experiment can be seen in Figure 7.6.

It can be seen that the presence of a body has a negative effect on the localization accuracy as expected. Both indoors and outdoors the average error is larger with a body present.

It is expected that with a body in place, signals bounce off the wall/floor before they arrive at the receiving device, which would mean that the signal has travelled longer. To verify this we look at the comparison of the data in Table D.2. Distances between device (4-2), (4-3) and (4-7) are likely to be affected by the human body because it is between these devices. When looking at these comparisons we see that all these combinations display a longer mean measured distance. To verify this, another experiment is performed. All but one devices are placed on a table, close together in a straight line. The other device is placed at 5 m distance. Data collection is started and after 250 measurements a body is placed directly in front of the single device and the experiment is continued.

The results can be seen in Figure 7.5 where it shows that distance is indeed overestimated when a body is present between the measuring devices.

The average estimated location with and without body also shows this effect. Average estimated location without a body is (4.88, 5.00) and with a body: (4.97, 4.88). This is consistent with a longer estimated distance between device (4-2), (4-3) and (4-7).

In Figure 7.6 the experiment when a person is walking through the playing field is also shown. A person that is present in the playing field and does not have the device close to the body does not have a significant effect on the location accuracy.

### 7.7 Indoors versus outdoors

The dataset in section 7.2 is from an indoors experiment. To compare indoors with outdoors performance the same experiment is performed outdoors. Devices are placed at the same relative locations and data is stored. The placement of the anchors and blind nodes however is much less

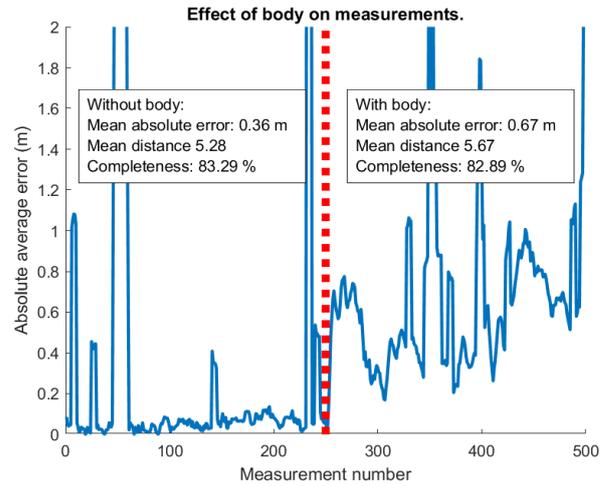


Figure 7.5: Effect of a body on the measurements.

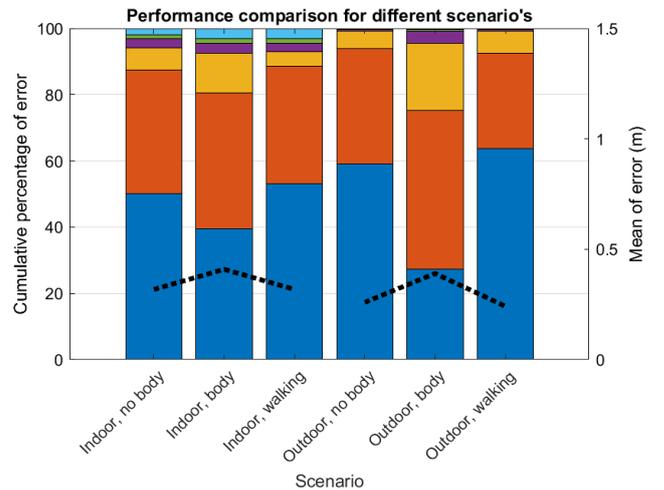


Figure 7.6: Effect of environmental variables. For the legend, see Figure 7.3.

accurate in an outdoors environment. In the indoors experiment the ground was painted with lines which can be used to precisely place the anchors. A comparison can be seen in Figure 7.6.

What can be seen is that the localization performance is comparable to the indoors environment. Outdoors there is less multipath interference since there are no walls for the UWB signal to bounce off. This would affect the localization accuracy, however UWB is very resistant to multipath interference. This results in comparable performance for in- and outdoors.

### 7.8 Effect completeness of dissimilarity matrix

The number of measured distances also has an effect on the performance of the localization. Matrices obtained in section 7.2 are complete to a certain degree. A 100 % complete matrix means that every device has measured the distance to all other devices in that superframe. From this measurement matrix a dissimilarity matrix is made which has a completeness that is the same or more than

that of the measurement matrix. Namely because a failed measurement from A to B can be replaced by a successful measurement from B to A.

We call the percentage of the distances that are in the dissimilarity matrix the completeness of the matrix. The effect of completeness of a matrix is plotted in Figure E.1c along with the completeness distribution. Dissimilarity matrices are separated into different sets based on this completeness and the mean error of this set is calculated. What can be seen is that the overall trend is that a fuller matrix results in a better location estimation as expected.

To validate this the large field outdoors dataset is used again. The first dissimilarity matrix is used to estimate a location and location estimation performs very poor:  $> 20 m$  error. This is not strange since completeness of the matrix is at a low 75%. In the next step, the first two dissimilarity matrices are joined: replacing missing entries and averaging double entries. This results in a completeness of the matrix of  $> 85\%$  and an error of  $1.5 m$ . These steps are repeated creating a more complete dissimilarity matrix with each step. The result is plotted in Figure E.1d where we can see that error is more or less the inverse of completeness.

## 8 Conclusion

The design steps taken in this paper have led to a working system, with it's limitations. The research done in section 4 have resulted in accurate enough distance measurements between devices. Which lies at the basis of a system like this. It was shown that  $T_{reply}$  has a significant impact on the accuracy of distance measurements due to crystal error and should therefore be as short as possible. To improve scalability and keep accuracy, the mesh was divided into groups where an entire group answers sequentially to a single initialization message. Also we found that the presence of a human body has a negative effect on the distance measurements. On these ranging messages, synchronization information is piggybacked to synchronize the system clocks of the devices. This is needed to make sure that each device performs the correct task in each timeslot: one device initiates distance measurements while the rest listens and answers. Timeslots are numbered and each device is appointed one according to it's ID. Shorter timeslots mean better scalability and/or better responsiveness. This because there are more timeslots available and thus more devices can perform distance measurements. Or the superframe can be shortened which increases the update rate. Due to the use of BLE advertising packets for communication of measurement results to a device running the localization algorithm, the length of these timeslots has to be very long. This limits the scalability and/or responsiveness of the system simply because time is wasted repeating the advertising packets to achieve an acceptable reliability of communication.

Calculating a position from the measured distances is done using the MDS algorithm to find a best fit and then mapping the calculated positions to the known position of the anchors using procrustes algorithm. No research is done on the effect of using different algorithms and it

is expected that performance can be boosted with other algorithms. Also the accuracy of the used algorithm can be increases by using data from previous localizations or from other sensors.

Performance of the system is evaluated for separate measurements and different configurations are compared. Localization performance is heavily affected by this configuration. Mainly by the number of nodes and by the ratio of  $range/fieldsize$ . From the latter we have seen that performance of localization drops significantly if this ratio is lower than 1. Accuracy can be increased by placing extra anchors on the sides of the field and by adding extra players.

Larger fields do not necessarily have a negative impact on the quality of a single distance measurement. However the number of successful measurements decreases as UWB packets are not received reliably, resulting in a distance matrix being less filled. This has a negative impact on accuracy of the localization algorithm as is shown in section 7.5. This means that with this amount of devices localization on a large field is not nearly accurate enough.

The experiments performed in this research are static and the historical data was not used as input for the next location estimation to keep experiments comparable. Using this data however increases accuracy of the location estimation and should be done for a Real Time Location System (RTLS).

## 9 Recommendations

In this section a few recommendations are given. Implementing these recommendations would (significantly) increase the performance of the system in terms of energy consumption, responsiveness and accuracy. However due to time limitations they have not been implemented. The first recommendation that should be implemented is the one described in section 9.2.1 since that would have the most significant impact on the performance. Further important improvements are the robustness and data communication as explained in section 9.5.2 and section 9.4.1 respectively.

### 9.1 Improvements in Localization

The localization algorithm used is not studied extensively. It also only uses data from the current measurement and discards historical data. As explained in section 6 using this historical data improves the performance of the localization algorithm. From historical data, the previous speed and acceleration can be determined which can be used to predict locations. These speed and acceleration estimates can be refined with data from for instance an IMU sensor to refine this data. With the previous position and an estimation of the movement, a precise prediction of the new position can be made which can be used as the initial guess in the localization algorithm.

Since the completeness of a measurement has a significant impact on the performance of the MDS algorithm care should be taken to add enough devices to the field.

Missing distances in the measurement can be replaced by calculated distances from the initial guess of the locations.

The localization algorithm sometimes estimates a location which is outside the playing field or far from the previous location. These anomalies can be filtered out to further improve performance.

## 9.2 Improvements in accuracy

### 9.2.1 Data communication

As shown in section 5.2.1 the BLE communication implemented performs poorly. This means that a lot of measured distances are lost which probably negatively impacts the performance of the localization algorithm. A first solution would be to store the measured distances locally and use a more reliable communications method to transfer these distances. This however means that the system can not be used as a Real Time Localization System. To solve this a hybrid of these two can be used: Do course real time localization with the received distances and refine later with the stored distances.

A better approach would be to improve the communications. Since it is not possible to have all devices communicate via a BLE connection except when multiple BLE sinks are deployed we could implement a similar technique as in section 5.4.

In this, synchronization data is eventually distributed over the entire mesh network which could include a data sink to tunnel data to the localization algorithm. The method used in section 5.4 however is too slow but serves mainly as an indication.

### 9.2.2 Temperature

The temperature of the DW1000 IC has an effect on the accuracy of the distance measurements due to various reasons. The temperature has an effect on the crystal frequency stability of the DW1000, it also has an effect on the antenna delay, as stated in [45]. In the 5 m measurements the estimated distance varies a lot. The Decawave has an internal sensor which can be used to measure the temperature of the IC to compensate for the different temperatures. The crystal frequency stability problem could also be mitigated by using a Temperature Compensated Crystal Oscillator (TCXO).

### 9.2.3 Crystal Drift

As seen in Figure 4.4 the crystal drift seems to have a predictable effect on the estimated distance. In this case only one experiment with the same conditions has been performed, but it is expected that the drift of a specific crystal is always the same. Except for variations with temperature and supply voltage but the DW1000 features sensors for these variables and therefore we can correct for this. This means that the range error can be predicted using information on  $T_{reply}$  which is known. We can correct for this in the localization algorithm. For instance if device 0 estimates the distance to device 10 which has a  $T_{reply}$  of for instance<sup>3</sup> 7 ms we can predict that the dis-

<sup>3</sup>This is not the real value, but used here for the sake of clarity.

tance measured is approximately 4.5 m too small. This can then be taken into account for the localization but needs device specific information and thus a lot of testing.

### 9.2.4 Ranging scheme

In Figure 4.6 the first and last message look similar. This similarity can be exploited to decrease the amount of messages used for distance estimation. If the last message, containing the three timestamps from the initializing device (or Tag in the figure) also serves as an initializing message for the next group of distance estimations one message can be eliminated. This would require a group identifier in the message, so devices A, B and C (Anchor in the figure) know whether to use the reception timestamp of this message as the first or last timestamp. It would mean that the message labelled “Final” would be used as the final message for device A, B and C, but would also serve as the “Poll” message to device D, E and F.

### 9.2.5 Code

The implementation of the distance estimation on the devices has not been optimized at this point. These optimizations include: using smaller number representations, calculating distances when all replies are received, only updating parts of buffers that need updating and more. When optimized,  $T_{reply}$  can most definitely be decreased. Decreasing  $T_{reply}$  means increasing accuracy, or scalability if it is chosen to not have three but four or more devices answer per group as explained in section 4.5.

### 9.2.6 Hardware

The communication between the nRF52832 and DW1000 is done via SPI. The DW1000 has a maximum SPI speed of 20 MHz [46]. The nRF52832 however has a maximum SPI speed of 8 MHz [47]. Receiving an initiation packet and sending the response requires 41 bytes to be written to/read from the DW1000 in 11 transactions. The time this takes can be reduced by a factor of 2.5 if a microcontroller capable of 20 MHz SPI is used. It is however not a very significant performance increase as it would probably only save something in the order of 10  $\mu$ s.

## 9.3 Improvements in range

### 9.3.1 Temperature

The temperature also influences the power of the transmitted signal as explained in [48]. The measured temperature can therefore also be used to keep the transmit power close to the maximum of  $-41.3$  dBm/MHz.

### 9.3.2 Reception

It is possible to add an RX amplifier. This would improve reception quality of messages and therefore the range of reliable communication.

## 9.4 Improvements in scalability

### 9.4.1 Data communication

At this point the measured distances are advertised in BLE packets which are received by the central device. A measured distance is placed in a particular position in the message, corresponding to the device that distance is measured to. Due to the limited size of these messages we can only accommodate 13 distances in the current format. In this section a few improvements are suggested. They are not implemented again due to time restrictions and the fact that data communication was a necessity rather than a field of interest for this research.

As explained in section 4.2 the usable range and thus the maximum estimated distance is not expected to be more than 40 *m*. This means that we could represent the integer part of a measured distance by 6 ( $2^6 - 1 = 63$ ) bits if we take a little safety margin. The TDMA timeslots are 50 *ms* which means that we have 20 available and thus can accommodate 20 devices in one superframe. Let's say we improve the code a lot, make these timeslots smaller and can now accommodate 50 devices. Then we also need 6 bits for a device ID which identifies the range measurement. We could split up the advertising packets in parts of 2 Bytes, leaving us with 13 sets of 2 bytes. From the 16 bits in a set we now have used 6 for the integer part of a distance and 6 for the device ID, leaving 4 for the decimal part of a distance. These 4 bits give a step size of 6.25 *cm* which is more than accurate enough. It is even possible to accommodate up to 128 devices if a bit of resolution for the decimal part is sacrificed. An example of how 16 bits would be used to represent a distance can be seen in Appendix C. This means that still 13 distances can be communicated only now the distances can be to any device in the range 0-63, not just 0-13.

### 9.4.2 Dual Buffer

The DW1000 IC features a double receive buffer. If this is enabled a new message can be received while the microcontroller is handling the current message. This can decrease the  $T_{reply}$  time since the second and third reply in the group can be sent earlier. This could improve accuracy or can increase the number of devices per group and therefore make a more scalable system.

### 9.4.3 Timeslot length

At this point the timeslot length is limited due to BLE communication. If this is changed to the UWB mesh communication solution or some other solution which does not require this long, the timeslot can be made shorter. The ranging exchange now takes less than 10 *ms* for 12 devices. If communication is done within an acceptable time timeslots can be made shorter. This would mean that all devices can be sampled multiple times each second increasing responsiveness. Or that more devices can be sampled, increasing scalability.

## 9.5 Improvements in Code

### 9.5.1 Energy Consumption

If a device is the initiating device in the current timeslot the nRF52832 constantly polls the DW1000's `System Event Status Register` to see if the `RXFCG` bit is set. If this bit is set the incoming message is handled. During other timeslots the nRF52832 waits for an interrupt and executes a cycle burning loop.

Both these actions can be adjusted to prolong battery life. Instead of a cycle burning loop, the microcontroller could be put in a power saving mode and be woken up by the interrupt. Care has to be taken that the time it takes to wake up the microcontroller does not cause the device to miss the  $T_{reply}$  deadline. During the device's timeslot polling could be changed to an interrupt based solution. Replacing polling with interrupts saves SPI transactions and thus power. The microcontroller could also be placed in a power saving mode, however due to the short periods of time this would not save as much power as during other timeslots.

### 9.5.2 Robustness

Almost no error handling was implemented. This can result in an unpredictable system. One example of this can be seen in Figure B.1 where device 2 fails to communicate in the 601 – 1000 measurement slice. At this point a red led is toggled to indicate a failure.

After this some crude error handling has been implemented to counter this. Errors are counted and if more than a certain amount of consecutive errors occur the device is reset. Successfully executed UWB ranging subtract one from the current count. This means that if a device has a few faults no action is taken, but if there are a lot of errors in a row the device resets itself.

## References

- [1] Mohammad Ghavami, Lachlan Michael, and Ryuji Kohno. *Ultra wideband signals and systems in communication engineering*. John Wiley & Sons, 2007.
- [2] Sinan Gezici, Zhi Tian, Georgios B Giannakis, Hisashi Kobayashi, Andreas F Molisch, H Vincent Poor, and Zafer Sahinoglu. Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. *IEEE signal processing magazine*, 22(4):70–84, 2005.
- [3] Terry Phebey. The Ubisense assembly control solution for BMW. In *RTLS in Manufacturing Workshop-RFID Journal Europe Live*, 2010. URL [/http://www.rfidjournal.net/masterPresentations/rfid\\_europe2010/np/chandler\\_phebey\\_nov2\\_500\\_rtlsManu.pdf](http://www.rfidjournal.net/masterPresentations/rfid_europe2010/np/chandler_phebey_nov2_500_rtlsManu.pdf). visited on 24/09/2018.
- [4] Lorenzo Mucchi, Federico Trippi, and Alessio Carpini. Ultra wide band real-time location system for cinematic survey in sports. In *Applied Sciences in Biomedical and Communication Technologies (ISABEL), 2010 3rd International Symposium on*, pages 1–6. IEEE, 2010.
- [5] Matteo Ridolfi, Stef Vandermeeren, Jense Defraye, Heidi Steendam, Joeri Gerlo, Dirk De Clercq, Jeroen Hoebeke, and Eli De Poorter. Experimental evaluation of uwb indoor positioning for sport postures. *Sensors*, 18(1):168, 2018.
- [6] J. A. Corrales, F. A. Candelas, and F. Torres. Hybrid tracking of human operators using imu/uwb data fusion by a kalman filter. In *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 193–200, March 2008. doi: /10.1145/1349822.1349848.
- [7] Robert J Fontana and Steven J Gunderson. Ultra-wideband precision asset location system. In *Ultra Wideband Systems and Technologies, 2002. Digest of Papers. 2002 IEEE Conference on*, pages 147–150. IEEE, 2002.
- [8] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6):1067–1080, Nov 2007. ISSN 1094-6977. doi: /10.1109/TSMCC.2007.905750.
- [9] Humatics. Humatics home page. URL [/https://www.humatics.com/](https://www.humatics.com/). visited on 24/09/2018.
- [10] Tracktio. uRTLS™: Indoor & outdoor location solution. URL [/http://tracktio.com/rtls/](http://tracktio.com/rtls/). visited on 24/09/2018.
- [11] Zebra. Dart ultra wideband (uwb) technology. URL [/https://www.zebra.com/us/en/products/location-technologies/dart-uwb.html](https://www.zebra.com/us/en/products/location-technologies/dart-uwb.html). visited on 24/09/2018.
- [12] STATSports. Statsports mapps. URL [/https://statsports.com/mapps-technology/](https://statsports.com/mapps-technology/). visited on 11/10/2018.
- [13] Jasper Gerth. Real time localization of athletes using ultra wide band using a mesh network.
- [14] IsoLynx. Isolynx homepage. URL [/http://www.finishlynx.com/isolynx/](http://www.finishlynx.com/isolynx/). visited on 11/10/2018.
- [15] Abdulrahman Alarifi, AbdulMalik S. Al-Salman, Mansour Alsaleh, Ahmad Alnafessah, Suheer Alhadhrami, Mai A. Al-Ammar, and Hend Suliman Al-Khalifa. Ultra wideband indoor positioning technologies: Analysis and recent advances †. In *Sensors*, 2016.
- [16] Suheer Alhadhrami, A Al-Salman, H Al-Khalifa, Abdulrahman Alarifi, Ahmad Alnafessah, Mansour Alsaleh, and M Al-Ammar. Ultra wideband positioning: An analytical study of emerging technologies. In *Proceedings of the Eighth International Conference on Sensor Technologies and Applications, SENSORCOMM*, pages 1–9, 2014.
- [17] Jia Wang, Asad Khalid Raja, and Zhibo Pang. Prototyping and experimental comparison of ir-uwb based high precision localization technologies. In *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, pages 1187–1192. IEEE, 2015.
- [18] M Segura, V Mut, and C Sisterna. Ultra wideband indoor navigation system. *IET Radar, Sonar & Navigation*, 6(5):402–411, 2012.
- [19] D. Niculescu and B. Nath. Ad hoc positioning system (aps). In *GLOBECOM'01. IEEE Global Telecommunications Conference (Cat. No.01CH37270)*, volume 5, pages 2926–2931 vol.5, Nov 2001. doi: /10.1109/GLOCOM.2001.965964.
- [20] A. A. Agashe, A. A. Agashe, and R. S. Patil. Evaluation of dv hop localization algorithm in wireless sensor networks. In *2012 International Conference on Advances in Mobile Network, Communication and Its Applications*, pages 79–82, Aug 2012. doi: /10.1109/MNCApps.2012.21.
- [21] Qicai Shi, S. Kyperountas, N. S. Correal, and Feng Niu. Performance analysis of relative location estimation for multihop wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):830–838, April 2005. ISSN 0733-8716. doi: /10.1109/JSAC.2005.843560.
- [22] Neal Patwari, Alfred O Hero, Matt Perkins, Neiyer S Correal, and Robert J O’dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on signal processing*, 51(8):2137–2148, 2003.

- [23] Neiyer S Correal, Spyros Kyperountas, Qicai Shi, and Matt Welborn. An uwb relative location system. In *IEEE Conference on Ultra Wideband Systems and Technologies, 2003*, pages 394–397. IEEE, 2003.
- [24] Norman Abramson. The aloha system: another alternative for computer communications. In *Proceedings of the November 17-19, 1970, fall joint computer conference*, pages 281–285. ACM, 1970.
- [25] Jose A Costa, Neal Patwari, and Alfred O Hero III. Distributed weighted-multidimensional scaling for node localization in sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 2(1):39–64, 2006.
- [26] Vo Dinh Minh Nhat, Duc Vo, Subhash Challa, and SungYoung Lee. Nonmetric mds for sensor localization. In *2008 3rd International Symposium on Wireless Pervasive Computing*, pages 396–400. IEEE, 2008.
- [27] pozyx. pozyx homepage. URL [/https://www.pozyx.io/](https://www.pozyx.io/). visited on 17/01/2019.
- [28] Decawave. Decawave partners website, . URL [/https://www.decawave.com/partners/](https://www.decawave.com/partners/). visited on 17/01/2019.
- [29] Decawave. DW1000 user manual v2.15, 09 2018. URL [/https://decawave.com/sites/default/files/dw100020user20manual\\_0.pdf](https://decawave.com/sites/default/files/dw100020user20manual_0.pdf).
- [30] Decawave. DW1000 datasheet v2.17, 08 2018. URL [/https://www.decawave.com/wp-content/uploads/2018/09/dw1000\\_datasheet\\_v2.17.pdf](https://www.decawave.com/wp-content/uploads/2018/09/dw1000_datasheet_v2.17.pdf).
- [31] Decawave. DWM1001 datasheet v1.10, 02 2018. URL [/https://www.decawave.com/wp-content/uploads/2018/08/dwm1001\\_datasheet.pdf](https://www.decawave.com/wp-content/uploads/2018/08/dwm1001_datasheet.pdf).
- [32] Decawave. DWM1001 schematic v1.2, 06 2018. URL [/https://www.decawave.com/wp-content/uploads/2018/08/dwm1001\\_module\\_schematics\\_vd1.2.pdf](https://www.decawave.com/wp-content/uploads/2018/08/dwm1001_module_schematics_vd1.2.pdf).
- [33] Decawave. Dwm1000 product brief v1.2, . URL [/https://www.decawave.com/wp-content/uploads/2018/09/dwm1000\\_brief.pdf](https://www.decawave.com/wp-content/uploads/2018/09/dwm1000_brief.pdf).
- [34] Decawave. APS003: REAL TIME LOCATION SYSTEMS v1.1, . URL [/https://www.decawave.com/wp-content/uploads/2018/10/APS003\\_DW1000-RTLS-Introduction\\_v1.1.pdf](https://www.decawave.com/wp-content/uploads/2018/10/APS003_DW1000-RTLS-Introduction_v1.1.pdf).
- [35] Michael McLaughlin and Billy Verso. Asymmetric double-sided two-way ranging in an ultrawideband communication system, 11 2016. US Patent App. 15/500,633.
- [36] Decawave. APS011: SOURCES OF ERROR IN DW1000 BASED TWO-WAY RANGING (TWR) SCHEMES v1.0, . URL [/https://www.decawave.com/wp-content/uploads/2018/08/aps011\\_sources\\_of\\_error\\_in\\_twr.pdf](https://www.decawave.com/wp-content/uploads/2018/08/aps011_sources_of_error_in_twr.pdf).
- [37] Guowang Miao, Jens Zander, Ki Won Sung, and Slimane Ben Slimane. *Fundamentals of mobile data networks*. Cambridge University Press, 2016.
- [38] Bluetooth Special Interest Group. *Bluetooth Core Specification v5.1*, 01 2019. URL [/https://www.bluetooth.com/specifications/bluetooth-core-specification](https://www.bluetooth.com/specifications/bluetooth-core-specification).
- [39] Bluetooth Special Interest Group. Bluetooth homepage. URL [/https://www.bluetooth.com/](https://www.bluetooth.com/). visited on 27/03/2019.
- [40] Kay Römer, Philipp Blum, and Lennart Meier. Time synchronization and calibration in wireless sensor networks. *Handbook of sensor networks: Algorithms and architectures*, pages 199–237, 2005.
- [41] Qun Li and Daniela Rus. Global clock synchronization in sensor networks. *IEEE Transactions on computers*.
- [42] Matlab. Help on nonclassical multidimensional scaling, 2019. URL [/https://nl.mathworks.com/help/stats/mdscale.html](https://nl.mathworks.com/help/stats/mdscale.html).
- [43] Neal Patwari, Robert J O’Dea, and Yanwei Wang. Relative location in wireless networks. In *IEEE VTS 53rd Vehicular Technology Conference, Spring 2001. Proceedings (Cat. No. 01CH37202)*, volume 2, pages 1149–1153. IEEE, 2001.
- [44] Neal Patwari, Alfred O Hero, Matt Perkins, Neiyer S Correal, and Robert J O’dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on signal processing*, 51(8):2137–2148, 2003.
- [45] APS014: ANTENNA DELAY CALIBRATION OF DW1000-BASED PRODUCTS AND SYSTEMS v1.01. URL [/https://www.decawave.com/wp-content/uploads/2018/08/aps014-antennadelaycalibrationofdw1000-basedproductsandsystems\\_v1.01.pdf](https://www.decawave.com/wp-content/uploads/2018/08/aps014-antennadelaycalibrationofdw1000-basedproductsandsystems_v1.01.pdf).
- [46] Decawave. Dw1000 twr calculator, . URL [/https://www.decawave.com/sites/default/files/dw1000\\_product\\_powercalc\\_ver1.5\\_0.zip](https://www.decawave.com/sites/default/files/dw1000_product_powercalc_ver1.5_0.zip). visited on 24/10/2018.
- [47] Nordic Semiconductor. nrf52832 product specification v1.4, 10 2017. URL [/https://www.nordicsemi.com/DocLib?Product=nRF52832](https://www.nordicsemi.com/DocLib?Product=nRF52832).
- [48] Decawave. APS023: DW1000 TX BANDWIDTH AND CHANNEL POWER COMPENSATION v1.1, 04 2017. URL [/https://www.decawave.com/wp-content/uploads/2018/08/aps023\\_part\\_2\\_tx\\_bandwidth\\_and\\_power\\_compensation\\_v1.1.pdf](https://www.decawave.com/wp-content/uploads/2018/08/aps023_part_2_tx_bandwidth_and_power_compensation_v1.1.pdf).

## A Table of measurements at different reply times

$T_{reply} \rightarrow$ Device $\downarrow$	1 ms	2 ms	3 ms	4 ms	5 ms	6 ms	7 ms	8 ms	9 ms	10 ms
1	$ \mu - 8 $ 0.564 m $\mu$ 7.436 m $\sigma$ 0.027 m	$ \mu - 8 $ 1.082 m $\mu$ 6.918 m $\sigma$ 0.051 m	$ \mu - 8 $ 1.597 m $\mu$ 6.403 m $\sigma$ 0.024 m	$ \mu - 8 $ 2.119 m $\mu$ 5.881 m $\sigma$ 0.025 m	$ \mu - 8 $ 2.650 m $\mu$ 5.350 m $\sigma$ 0.033 m	$ \mu - 8 $ 3.189 m $\mu$ 4.811 m $\sigma$ 0.029 m	$ \mu - 8 $ 3.696 m $\mu$ 4.304 m $\sigma$ 0.346 m	$ \mu - 8 $ 4.391 m $\mu$ 3.609 m $\sigma$ 0.036 m	$ \mu - 8 $ 5.095 m $\mu$ 2.905 m $\sigma$ 0.094 m	$ \mu - 8 $ 5.587 m $\mu$ 2.413 m $\sigma$ 0.043 m
2	$ \mu - 8 $ 0.239 m $\mu$ 7.761 m $\sigma$ 0.038 m	$ \mu - 8 $ 0.276 m $\mu$ 7.724 m $\sigma$ 0.046 m	$ \mu - 8 $ 0.471 m $\mu$ 7.529 m $\sigma$ 0.073 m	$ \mu - 8 $ 0.600 m $\mu$ 7.400 m $\sigma$ 0.120 m	$ \mu - 8 $ 0.597 m $\mu$ 7.403 m $\sigma$ 0.087 m	$ \mu - 8 $ 0.803 m $\mu$ 7.197 m $\sigma$ 0.272 m	$ \mu - 8 $ 0.858 m $\mu$ 7.142 m $\sigma$ 0.273 m	$ \mu - 8 $ 1.136 m $\mu$ 6.864 m $\sigma$ 0.254 m	$ \mu - 8 $ 1.375 m $\mu$ 6.625 m $\sigma$ 0.195 m	$ \mu - 8 $ 1.688 m $\mu$ 6.312 m $\sigma$ 0.190 m
3	$ \mu - 8 $ 0.351 m $\mu$ 7.649 m $\sigma$ 0.031 m	$ \mu - 8 $ 0.587 m $\mu$ 7.413 m $\sigma$ 0.028 m	$ \mu - 8 $ 0.815 m $\mu$ 7.185 m $\sigma$ 0.078 m	$ \mu - 8 $ 1.066 m $\mu$ 6.934 m $\sigma$ 0.062 m	$ \mu - 8 $ 1.316 m $\mu$ 6.684 m $\sigma$ 0.031 m	$ \mu - 8 $ 1.566 m $\mu$ 6.434 m $\sigma$ 0.034 m	$ \mu - 8 $ 1.827 m $\mu$ 6.173 m $\sigma$ 0.072 m	$ \mu - 8 $ 2.113 m $\mu$ 5.887 m $\sigma$ 0.037 m	$ \mu - 8 $ 2.441 m $\mu$ 5.559 m $\sigma$ 0.041 m	$ \mu - 8 $ 2.720 m $\mu$ 5.280 m $\sigma$ 0.039 m
4	$ \mu - 8 $ 0.324 m $\mu$ 7.676 m $\sigma$ 0.025 m	$ \mu - 8 $ 0.597 m $\mu$ 7.403 m $\sigma$ 0.052 m	$ \mu - 8 $ 1.127 m $\mu$ 6.873 m $\sigma$ 0.104 m	$ \mu - 8 $ 1.362 m $\mu$ 6.638 m $\sigma$ 0.040 m	$ \mu - 8 $ 1.693 m $\mu$ 6.307 m $\sigma$ 0.051 m	$ \mu - 8 $ 2.062 m $\mu$ 5.938 m $\sigma$ 0.125 m	$ \mu - 8 $ 2.597 m $\mu$ 5.403 m $\sigma$ 0.150 m	$ \mu - 8 $ 3.215 m $\mu$ 4.785 m $\sigma$ 0.090 m	$ \mu - 8 $ 3.649 m $\mu$ 4.351 m $\sigma$ 0.071 m	$ \mu - 8 $ 4.142 m $\mu$ 3.858 m $\sigma$ 0.185 m
5	$ \mu - 8 $ 0.421 m $\mu$ 7.579 m $\sigma$ 0.036 m	$ \mu - 8 $ 0.736 m $\mu$ 7.264 m $\sigma$ 0.070 m	$ \mu - 8 $ 1.068 m $\mu$ 6.932 m $\sigma$ 0.158 m	$ \mu - 8 $ 1.267 m $\mu$ 6.733 m $\sigma$ 0.066 m	$ \mu - 8 $ 1.693 m $\mu$ 6.307 m $\sigma$ 0.113 m	$ \mu - 8 $ 1.803 m $\mu$ 6.197 m $\sigma$ 0.331 m	$ \mu - 8 $ 2.522 m $\mu$ 5.478 m $\sigma$ 0.137 m	$ \mu - 8 $ 3.116 m $\mu$ 4.884 m $\sigma$ 0.283 m	$ \mu - 8 $ 3.573 m $\mu$ 4.427 m $\sigma$ 0.149 m	$ \mu - 8 $ 4.248 m $\mu$ 3.752 m $\sigma$ 0.166 m
6	$ \mu - 8 $ 0.407 m $\mu$ 7.593 m $\sigma$ 0.058 m	$ \mu - 8 $ 0.631 m $\mu$ 7.369 m $\sigma$ 0.074 m	$ \mu - 8 $ 0.868 m $\mu$ 7.132 m $\sigma$ 0.332 m	$ \mu - 8 $ 1.054 m $\mu$ 6.946 m $\sigma$ 0.300 m	$ \mu - 8 $ 1.382 m $\mu$ 6.618 m $\sigma$ 0.151 m	$ \mu - 8 $ 1.773 m $\mu$ 6.227 m $\sigma$ 0.206 m	$ \mu - 8 $ 1.975 m $\mu$ 6.025 m $\sigma$ 0.312 m	$ \mu - 8 $ 2.084 m $\mu$ 5.916 m $\sigma$ 0.340 m	$ \mu - 8 $ 2.510 m $\mu$ 5.490 m $\sigma$ 0.367 m	$ \mu - 8 $ 3.078 m $\mu$ 4.922 m $\sigma$ 0.420 m
7	$ \mu - 8 $ 0.314 m $\mu$ 7.686 m $\sigma$ 0.029 m	$ \mu - 8 $ 0.566 m $\mu$ 7.434 m $\sigma$ 0.030 m	$ \mu - 8 $ 0.818 m $\mu$ 7.182 m $\sigma$ 0.033 m	$ \mu - 8 $ 1.089 m $\mu$ 6.911 m $\sigma$ 0.051 m	$ \mu - 8 $ 1.366 m $\mu$ 6.634 m $\sigma$ 0.028 m	$ \mu - 8 $ 1.661 m $\mu$ 6.339 m $\sigma$ 0.028 m	$ \mu - 8 $ 1.918 m $\mu$ 6.082 m $\sigma$ 0.292 m	$ \mu - 8 $ 2.383 m $\mu$ 5.617 m $\sigma$ 0.035 m	$ \mu - 8 $ 2.760 m $\mu$ 5.240 m $\sigma$ 0.048 m	$ \mu - 8 $ 3.321 m $\mu$ 4.679 m $\sigma$ 0.035 m
8	$ \mu - 8 $ 0.381 m $\mu$ 7.619 m $\sigma$ 0.036 m	$ \mu - 8 $ 0.851 m $\mu$ 7.149 m $\sigma$ 0.305 m	$ \mu - 8 $ 0.765 m $\mu$ 7.235 m $\sigma$ 0.200 m	$ \mu - 8 $ 0.981 m $\mu$ 7.019 m $\sigma$ 0.259 m	$ \mu - 8 $ 1.312 m $\mu$ 6.688 m $\sigma$ 0.103 m	$ \mu - 8 $ 1.588 m $\mu$ 6.412 m $\sigma$ 0.113 m	$ \mu - 8 $ 1.773 m $\mu$ 6.227 m $\sigma$ 0.062 m	$ \mu - 8 $ 1.987 m $\mu$ 6.013 m $\sigma$ 0.288 m	$ \mu - 8 $ 2.274 m $\mu$ 5.726 m $\sigma$ 0.049 m	$ \mu - 8 $ 2.548 m $\mu$ 5.452 m $\sigma$ 0.050 m
9	$ \mu - 8 $ 0.046 m $\mu$ 7.954 m $\sigma$ 0.193 m	$ \mu - 8 $ 0.010 m $\mu$ 8.010 m $\sigma$ 0.259 m	$ \mu - 8 $ 0.100 m $\mu$ 8.100 m $\sigma$ 0.298 m	$ \mu - 8 $ 0.273 m $\mu$ 8.273 m $\sigma$ 0.319 m	$ \mu - 8 $ 0.317 m $\mu$ 8.317 m $\sigma$ 0.311 m	$ \mu - 8 $ 0.319 m $\mu$ 8.319 m $\sigma$ 0.330 m	$ \mu - 8 $ 0.309 m $\mu$ 8.309 m $\sigma$ 0.339 m	$ \mu - 8 $ 0.035 m $\mu$ 7.965 m $\sigma$ 0.209 m	$ \mu - 8 $ 0.217 m $\mu$ 7.783 m $\sigma$ 0.056 m	$ \mu - 8 $ 0.412 m $\mu$ 7.588 m $\sigma$ 0.046 m
10	$ \mu - 8 $ 0.790 m $\mu$ 7.210 m $\sigma$ 0.023 m	$ \mu - 8 $ 1.457 m $\mu$ 6.543 m $\sigma$ 0.024 m	$ \mu - 8 $ 2.139 m $\mu$ 5.861 m $\sigma$ 0.038 m	$ \mu - 8 $ 2.780 m $\mu$ 5.220 m $\sigma$ 0.030 m	$ \mu - 8 $ 3.457 m $\mu$ 4.543 m $\sigma$ 0.037 m	$ \mu - 8 $ 4.220 m $\mu$ 3.780 m $\sigma$ 0.147 m	$ \mu - 8 $ 4.561 m $\mu$ 3.439 m $\sigma$ 0.307 m	$ \mu - 8 $ 5.703 m $\mu$ 2.297 m $\sigma$ 0.060 m	$ \mu - 8 $ 6.551 m $\mu$ 1.449 m $\sigma$ 0.029 m	$ \mu - 8 $ 7.385 m $\mu$ 0.615 m $\sigma$ 0.035 m
11	$ \mu - 8 $ 0.366 m $\mu$ 7.634 m $\sigma$ 0.032 m	$ \mu - 8 $ 0.636 m $\mu$ 7.364 m $\sigma$ 0.030 m	$ \mu - 8 $ 0.556 m $\mu$ 7.444 m $\sigma$ 0.333 m	$ \mu - 8 $ 1.155 m $\mu$ 6.845 m $\sigma$ 0.032 m	$ \mu - 8 $ 1.431 m $\mu$ 6.569 m $\sigma$ 0.040 m	$ \mu - 8 $ 1.690 m $\mu$ 6.310 m $\sigma$ 0.036 m	$ \mu - 8 $ 2.038 m $\mu$ 5.962 m $\sigma$ 0.268 m	$ \mu - 8 $ 2.407 m $\mu$ 5.593 m $\sigma$ 0.051 m	$ \mu - 8 $ 2.761 m $\mu$ 5.239 m $\sigma$ 0.044 m	$ \mu - 8 $ 3.189 m $\mu$ 4.811 m $\sigma$ 0.053 m

Table A.1: Effect of  $T_{reply}$  on ranging accuracy. Distance measurements are performed at 8.00 m . Errors in measurement larger than 1 m and standard deviations larger than 0.20 m are colored red.

## B Quality of synchronization

Here the quality of synchronization for devices 0-3 are shown. Plotted is the mean and standard deviation of the edges indicating the timeslot. This is the same as the experiment in Figure 5.2 but shown here for all devices and for a longer time. The experiment was started and any anomalies that were seen were written down with the approximate corresponding measurement slice. After 3000 measurements are recorded, the measurements in which not all 4 devices provide a rising and falling edge are deleted, a total of 33 are deleted. This means that the measurements after a fail are shifted 1 position to the left on the x-axis in the figures.

What immediately stands out is the high deviation from ideal of device 2. In the time slice 601 – 1000 the mean is very far below the ideal value with a huge standard deviation. In this timeslot device 2 was in a fail state at slice 750 – 880 and therefore did not send or receive any UWB messages. This means that the device did not synchronize with the rest of the mesh network at this time. Also in slice 2360–2460 device 2 was in a fail state.

Device 3 has higher deviations from ideal from the 1401 – 1600 timeslot and later. A 1500 device 6 was reintroduced in the mesh. This was removed since it did not work properly and reintroduced for testing purposes. Apparently device 3 was influenced more by this device.

The mean and standard deviation of device 0 is very small due to this device triggering the measurement. This means that the rising edge is always perfectly timed since this is device as time zero.

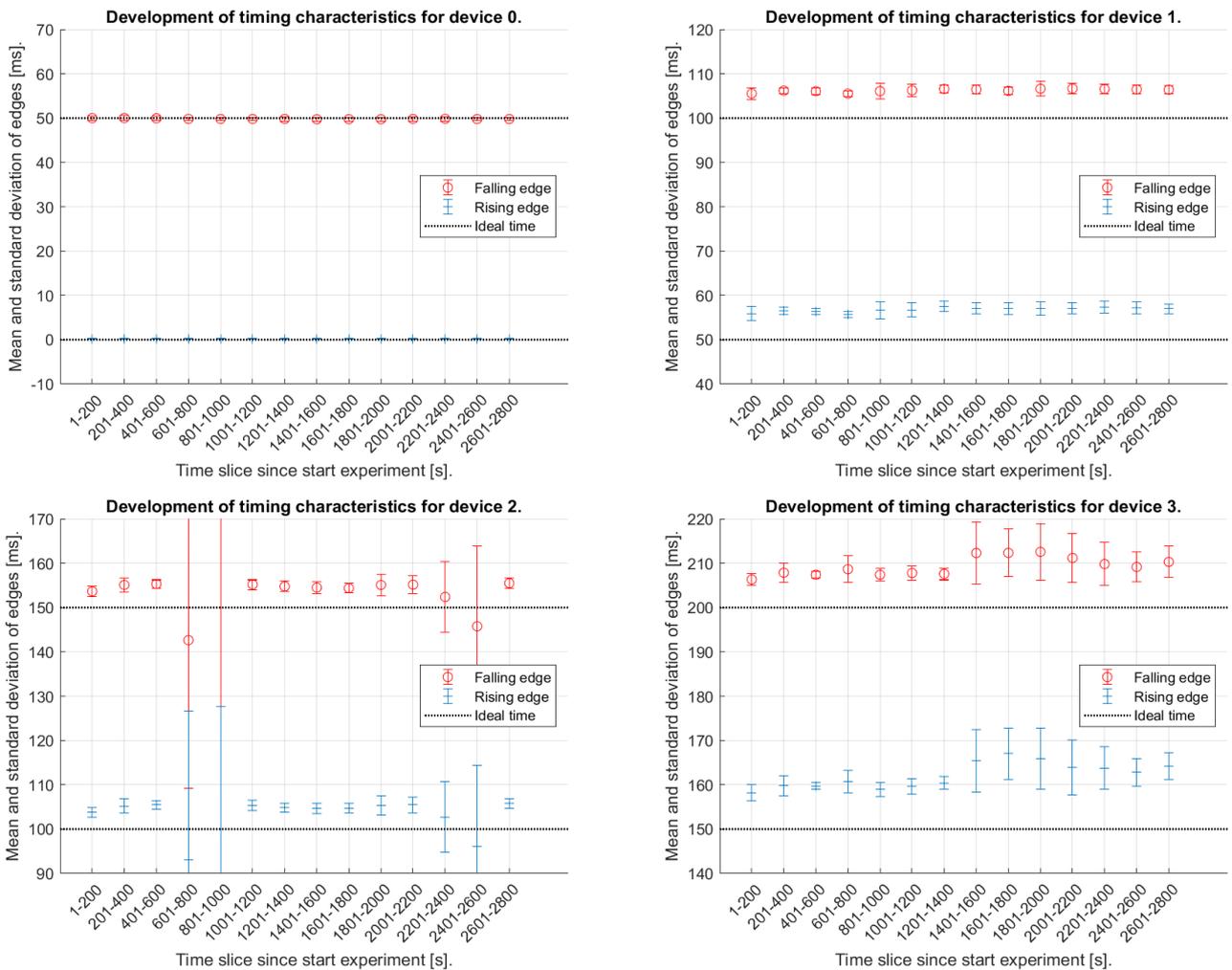


Figure B.1: Development of the accuracy of synchronization over time for extended time with devices 0-11 powered on.

### B.1 Deviation from superframe

A measurement has been done on the accuracy of the superframe timing. In this experiment only device 0 is hooked up to the oscilloscope and a longer timebase is used, again device 0-3 are powered on. Two consecutive rising edges are now captured, indicating how long a second takes in device time on device 0. Again the progression is shown during the experiment in Figure B.2. From now on the term 'real second' refers to a second measured on the oscilloscope, which is calibrated and very precise.

What stands out is that the second measured in system time is always shorter than a real second. This behaviour is also seen in the software of the device. If we use the debugging capabilities we can read the value of the time correction (see section 5.3) that is calculated without interfering with the device performance. The experiment is performed, and the results are plotted in Figure B.3. What can be seen is that this correction is not exactly the same as the deviation from the real second in Figure B.2 but it is in the same order of magnitude. This possibly enables us to include a device specific correction,

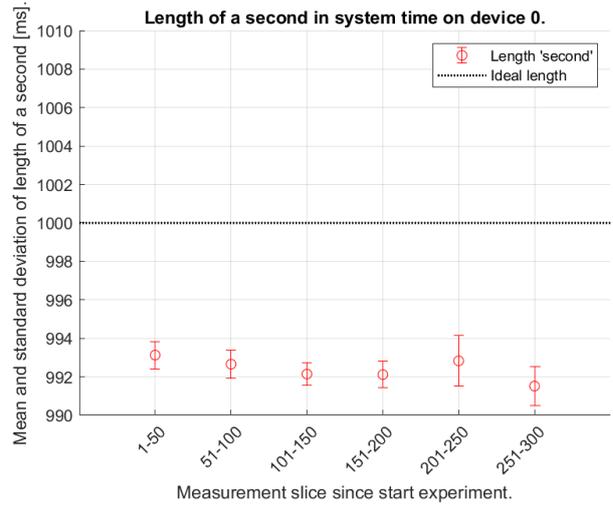


Figure B.2: Length of a second in corrected time on device 0.

which should increase the quality of synchronization.

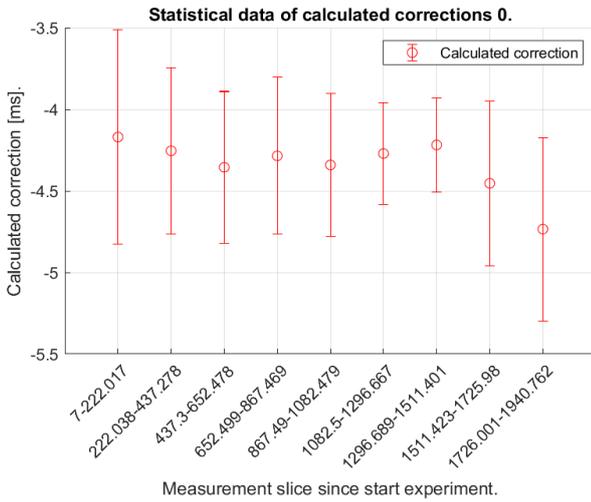


Figure B.3: Calculated corrections on device 0.

## C Proposal for message format in BLE communication

If the advertising packets from BLE are divided in sets of 2 bytes these two bytes can be used to represent a distance measurement along with the corresponding device ID. The representation is shown here in Table C.1. In the table an example is provided. A distance measurement of 21.81 *m* to device 39.

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Content	Device ID 0-63						Integer part 0-63					Decimal part 0 - 0.9375				
Example	1	0	0	1	1	1	0	1	0	1	0	1	1	1	0	1
Decoded	39						21					$13 \cdot (1/16) = 0.8125$				

Table C.1: 16 bit representation of a distance measurement to a device.

If more devices are needed, one bit of accuracy can be sacrificed. An example can be seen in Table C.2 where a distance of 44.63 to device 92 is shown.

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Content	Device ID 0-127							Integer part 0-63					Decimal part 0 - 0.875			
Example	1	0	1	1	1	0	0	1	0	1	1	0	0	1	0	1
Decoded	92							44					$5 \cdot (1/8) = 0.625$			

Table C.2: 16 bit representation of a distance measurement to a device when more devices are needed.

## D Quality of experiment data

To → From ↓	0	1	2	3	4	5	6	7	8	9
0	-	<b>Success rate 82.4 %</b> $\mu$ Measured 9.9 m <b><math>\sigma</math> Measured 2.48 m</b> Real 10.0 m <b>Error 1.3 %</b>	Success rate 86.4 % $\mu$ Measured 13.0 m $\sigma$ Measured 0.23 m Real 14.1 m Error 8.1 %	Success rate 86.2 % $\mu$ Measured 10.0 m $\sigma$ Measured 0.10 m Real 10.0 m <b>Error 0.1 %</b>	<b>Success rate 81.0 %</b> $\mu$ Measured 6.8 m $\sigma$ Measured 0.18 m Real 7.1 m <b>Error 4.3 %</b>	<b>Success rate 75.0 %</b> $\mu$ Measured 4.1 m $\sigma$ Measured 0.28 m Real 4.6 m <b>Error 11.6 %</b>	Success rate 85.8 % $\mu$ Measured 8.3 m $\sigma$ Measured 0.10 m Real 8.5 m <b>Error 1.8 %</b>	Success rate 85.4 % $\mu$ Measured 7.3 m $\sigma$ Measured 0.16 m Real 7.3 m <b>Error 0.2 %</b>	Success rate 86.2 % $\mu$ Measured 6.7 m $\sigma$ Measured 0.25 m Real 6.8 m <b>Error 2.0 %</b>	Success rate 87.2 % $\mu$ Measured 6.4 m $\sigma$ Measured 0.09 m Real 6.6 m <b>Error 2.6 %</b>
1	<b>Success rate 91.8 %</b> $\mu$ Measured 10.3 m <b><math>\sigma</math> Measured 3.26 m</b> Real 10.0 m <b>Error 3.0 %</b>	-	<b>Success rate 91.4 %</b> $\mu$ Measured 9.2 m $\sigma$ Measured 0.04 m Real 10.0 m Error 8.4 %	<b>Success rate 91.6 %</b> $\mu$ Measured 14.1 m <b><math>\sigma</math> Measured 0.02 m</b> Real 14.1 m <b>Error 0.3 %</b>	Success rate 85.0 % $\mu$ Measured 7.0 m $\sigma$ Measured 0.03 m Real 7.1 m <b>Error 0.6 %</b>	<b>Success rate 77.2 %</b> $\mu$ Measured 5.4 m $\sigma$ Measured 0.04 m Real 5.4 m <b>Error 0.6 %</b>	<b>Success rate 90.2 %</b> $\mu$ Measured 2.6 m <b><math>\sigma</math> Measured 0.02 m</b> Real 2.8 m Error 6.7 %	Success rate 89.8 % $\mu$ Measured 8.0 m <b><math>\sigma</math> Measured 0.02 m</b> Real 7.7 m <b>Error 3.8 %</b>	<b>Success rate 90.8 %</b> $\mu$ Measured 7.1 m <b><math>\sigma</math> Measured 0.02 m</b> Real 6.8 m <b>Error 4.6 %</b>	<b>Success rate 91.4 %</b> $\mu$ Measured 3.5 m <b><math>\sigma</math> Measured 0.02 m</b> Real 3.4 m <b>Error 2.2 %</b>
2	Success rate 89.6 % $\mu$ Measured 14.3 m $\sigma$ Measured 0.09 m Real 14.1 m <b>Error 1.4 %</b>	<b>Success rate 0.0 %</b> $\mu$ Measured - $\sigma$ Measured - Real 10.0 m Error -	-	Success rate 90.0 % $\mu$ Measured 10.2 m <b><math>\sigma</math> Measured 0.02 m</b> Real 10.0 m <b>Error 2.5 %</b>	<b>Success rate 83.4 %</b> $\mu$ Measured 7.6 m $\sigma$ Measured 0.04 m Real 7.1 m Error 6.8 %	<b>Success rate 73.0 %</b> $\mu$ Measured 11.7 m $\sigma$ Measured 0.06 m Real 11.3 m <b>Error 3.7 %</b>	Success rate 88.4 % $\mu$ Measured 8.2 m <b><math>\sigma</math> Measured 0.02 m</b> Real 8.1 m <b>Error 1.4 %</b>	Success rate 88.6 % $\mu$ Measured 7.7 m $\sigma$ Measured 0.03 m Real 6.9 m <b>Error 11.2 %</b>	Success rate 89.4 % $\mu$ Measured 8.4 m $\sigma$ Measured 0.05 m Real 7.4 m <b>Error 14.6 %</b>	Success rate 90.0 % $\mu$ Measured 10.6 m <b><math>\sigma</math> Measured 0.02 m</b> Real 10.5 m <b>Error 1.2 %</b>
3	<b>Success rate 90.8 %</b> $\mu$ Measured 10.1 m $\sigma$ Measured 0.10 m Real 10.0 m <b>Error 0.7 %</b>	<b>Success rate 91.8 %</b> $\mu$ Measured 13.9 m <b><math>\sigma</math> Measured 0.02 m</b> Real 14.1 m <b>Error 2.0 %</b>	<b>Success rate 92.2 %</b> $\mu$ Measured 9.0 m $\sigma$ Measured 0.04 m Real 10.0 m Error 9.6 %	-	<b>Success rate 83.8 %</b> $\mu$ Measured 6.9 m $\sigma$ Measured 0.03 m Real 7.1 m <b>Error 2.1 %</b>	<b>Success rate 73.6 %</b> $\mu$ Measured 10.6 m $\sigma$ Measured 0.04 m Real 11.0 m <b>Error 3.0 %</b>	Success rate 90.0 % $\mu$ Measured 11.4 m <b><math>\sigma</math> Measured 0.02 m</b> Real 11.4 m <b>Error 0.1 %</b>	<b>Success rate 90.4 %</b> $\mu$ Measured 6.5 m <b><math>\sigma</math> Measured 0.02 m</b> Real 6.4 m <b>Error 0.2 %</b>	<b>Success rate 90.8 %</b> $\mu$ Measured 7.4 m <b><math>\sigma</math> Measured 0.03 m</b> Real 7.4 m <b>Error 0.5 %</b>	<b>Success rate 91.4 %</b> $\mu$ Measured 11.9 m <b><math>\sigma</math> Measured 0.02 m</b> Real 11.9 m <b>Error 0.5 %</b>
4	Success rate 87.8 % $\mu$ Measured 7.4 m <b><math>\sigma</math> Measured 3.84 m</b> Real 7.1 m <b>Error 4.3 %</b>	Success rate 88.8 % $\mu$ Measured 6.9 m $\sigma$ Measured 0.03 m Real 7.1 m <b>Error 2.3 %</b>	Success rate 89.0 % $\mu$ Measured 6.3 m $\sigma$ Measured 0.05 m Real 7.1 m <b>Error 10.8 %</b>	Success rate 85.0 % $\mu$ Measured 7.1 m <b><math>\sigma</math> Measured 0.02 m</b> Real 7.1 m <b>Error 0.1 %</b>	-	<b>Success rate 67.6 %</b> $\mu$ Measured 4.6 m $\sigma$ Measured 0.04 m Real 5.0 m Error 6.6 %	Success rate 86.8 % $\mu$ Measured 4.3 m $\sigma$ Measured 0.03 m Real 4.3 m <b>Error 0.9 %</b>	Success rate 86.4 % $\mu$ Measured 0.7 m $\sigma$ Measured 0.04 m Real 0.7 m <b>Error 11.0 %</b>	Success rate 86.2 % $\mu$ Measured 0.5 m $\sigma$ Measured 0.04 m Real 0.4 m <b>Error 21.1 %</b>	Success rate 88.0 % $\mu$ Measured 5.1 m $\sigma$ Measured 0.03 m Real 5.2 m <b>Error 1.9 %</b>
5	Success rate 85.6 % $\mu$ Measured 5.1 m <b><math>\sigma</math> Measured 4.51 m</b> Real 4.6 m <b>Error 10.3 %</b>	Success rate 87.0 % $\mu$ Measured 5.5 m $\sigma$ Measured 0.03 m Real 5.4 m <b>Error 1.9 %</b>	Success rate 87.2 % $\mu$ Measured 11.0 m $\sigma$ Measured 0.06 m Real 11.3 m Error 7.0 %	Success rate 87.4 % $\mu$ Measured 10.0 m <b><math>\sigma</math> Measured 0.02 m</b> Real 11.0 m <b>Error 0.3 %</b>	Success rate 87.0 % $\mu$ Measured 4.9 m $\sigma$ Measured 0.04 m Real 5.0 m <b>Error 2.2 %</b>	-	Success rate 85.4 % $\mu$ Measured 4.2 m <b><math>\sigma</math> Measured 0.02 m</b> Real 4.1 m <b>Error 1.5 %</b>	<b>Success rate 84.4 %</b> $\mu$ Measured 5.9 m $\sigma$ Measured 0.03 m Real 5.6 m Error 6.5 %	<b>Success rate 84.2 %</b> $\mu$ Measured 4.8 m $\sigma$ Measured 0.04 m Real 4.6 m <b>Error 4.0 %</b>	Success rate 86.2 % $\mu$ Measured 1.9 m $\sigma$ Measured 0.03 m Real 2.0 m Error 5.8 %
6	Success rate 88.0 % $\mu$ Measured 8.5 m <b><math>\sigma</math> Measured 1.97 m</b> Real 8.5 m <b>Error 1.0 %</b>	<b>Success rate 90.4 %</b> $\mu$ Measured 2.5 m $\sigma$ Measured 0.03 m Real 2.8 m <b>Error 11.1 %</b>	<b>Success rate 90.6 %</b> $\mu$ Measured 7.2 m $\sigma$ Measured 0.04 m Real 8.1 m <b>Error 11.7 %</b>	<b>Success rate 90.6 %</b> $\mu$ Measured 11.5 m <b><math>\sigma</math> Measured 0.02 m</b> Real 11.4 m <b>Error 0.6 %</b>	Success rate 89.2 % $\mu$ Measured 4.2 m <b><math>\sigma</math> Measured 0.02 m</b> Real 4.3 m <b>Error 1.7 %</b>	<b>Success rate 90.4 %</b> $\mu$ Measured 4.0 m $\sigma$ Measured 0.05 m Real 4.1 m <b>Error 3.7 %</b>	-	Success rate 87.4 % $\mu$ Measured 5.1 m <b><math>\sigma</math> Measured 0.02 m</b> Real 4.9 m <b>Error 3.9 %</b>	Success rate 87.8 % $\mu$ Measured 4.3 m $\sigma$ Measured 0.03 m Real 4.1 m Error 6.1 %	Success rate 88.4 % $\mu$ Measured 2.5 m $\sigma$ Measured 0.03 m Real 2.6 m <b>Error 2.3 %</b>
7	Success rate 87.6 % $\mu$ Measured 7.1 m $\sigma$ Measured 0.11 m Real 7.3 m <b>Error 2.5 %</b>	Success rate 90.0 % $\mu$ Measured 7.4 m <b><math>\sigma</math> Measured 0.02 m</b> Real 7.7 m <b>Error 4.6 %</b>	<b>Success rate 90.2 %</b> $\mu$ Measured 5.8 m $\sigma$ Measured 0.04 m Real 6.9 m <b>Error 16.2 %</b>	<b>Success rate 90.2 %</b> $\mu$ Measured 6.2 m <b><math>\sigma</math> Measured 0.02 m</b> Real 6.4 m <b>Error 3.9 %</b>	Success rate 87.8 % $\mu$ Measured 0.2 m $\sigma$ Measured 0.03 m Real 0.7 m <b>Error 67.3 %</b>	Success rate 87.4 % $\mu$ Measured 5.0 m $\sigma$ Measured 0.04 m Real 5.6 m Error 9.1 %	Success rate 89.6 % $\mu$ Measured 4.8 m <b><math>\sigma</math> Measured 0.02 m</b> Real 4.9 m <b>Error 3.7 %</b>	-	Success rate 86.2 % $\mu$ Measured 0.8 m $\sigma$ Measured 0.04 m Real 1.0 m <b>Error 24.8 %</b>	Success rate 87.6 % $\mu$ Measured 5.7 m $\sigma$ Measured 0.03 m Real 5.9 m <b>Error 2.8 %</b>
8	Success rate 86.0 % $\mu$ Measured 6.6 m <b><math>\sigma</math> Measured 1.92 m</b> Real 6.8 m <b>Error 3.4 %</b>	Success rate 88.0 % $\mu$ Measured 6.4 m <b><math>\sigma</math> Measured 0.02 m</b> Real 6.8 m Error 6.4 %	Success rate 88.2 % $\mu$ Measured 6.3 m $\sigma$ Measured 0.05 m Real 7.4 m <b>Error 15.0 %</b>	Success rate 88.4 % $\mu$ Measured 7.1 m $\sigma$ Measured 0.03 m Real 7.4 m <b>Error 3.8 %</b>	<b>Success rate 0.0 %</b> $\mu$ Measured - $\sigma$ Measured - Real 0.4 m Error -	<b>Success rate 3.8 %</b> $\mu$ Measured 3.8 m $\sigma$ Measured 0.05 m Real 4.6 m <b>Error 17.3 %</b>	<b>Success rate 57.2 %</b> $\mu$ Measured 3.8 m $\sigma$ Measured 0.31 m Real 4.1 m Error 5.6 %	<b>Success rate 58.2 %</b> $\mu$ Measured 0.9 m $\sigma$ Measured 0.04 m Real 1.0 m <b>Error 18.1 %</b>	-	<b>Success rate 70.4 %</b> $\mu$ Measured 4.5 m $\sigma$ Measured 0.29 m Real 4.8 m Error 6.1 %
9	Success rate 87.0 % $\mu$ Measured 7.2 m <b><math>\sigma</math> Measured 5.72 m</b> Real 6.6 m Error 9.6 %	Success rate 87.0 % $\mu$ Measured 3.1 m <b><math>\sigma</math> Measured 0.02 m</b> Real 3.4 m Error 8.4 %	<b>Success rate 90.8 %</b> $\mu$ Measured 9.3 m $\sigma$ Measured 0.04 m Real 10.5 m <b>Error 11.9 %</b>	<b>Success rate 91.2 %</b> $\mu$ Measured 11.8 m <b><math>\sigma</math> Measured 0.02 m</b> Real 11.9 m <b>Error 1.2 %</b>	Success rate 86.2 % $\mu$ Measured 4.8 m $\sigma$ Measured 0.03 m Real 5.2 m Error 7.2 %	<b>Success rate 81.6 %</b> $\mu$ Measured 1.4 m $\sigma$ Measured 0.05 m Real 2.0 m <b>Error 31.7 %</b>	<b>Success rate 90.2 %</b> $\mu$ Measured 2.4 m <b><math>\sigma</math> Measured 0.02 m</b> Real 2.6 m Error 8.8 %	<b>Success rate 91.0 %</b> $\mu$ Measured 5.8 m $\sigma$ Measured 0.03 m Real 5.9 m <b>Error 0.6 %</b>	Success rate 87.0 % $\mu$ Measured 4.7 m $\sigma$ Measured 0.03 m Real 4.8 m <b>Error 2.9 %</b>	-

Table D.1: Quality of the indoors without human body experiment data. Success rates lower than 85 %, a standard deviation of more than 0.50 m and an error of more than 10 % are colored red. Success rates higher than 90 %, a standard deviation of less than 0.03 m and an error of less than 5 % are colored green.

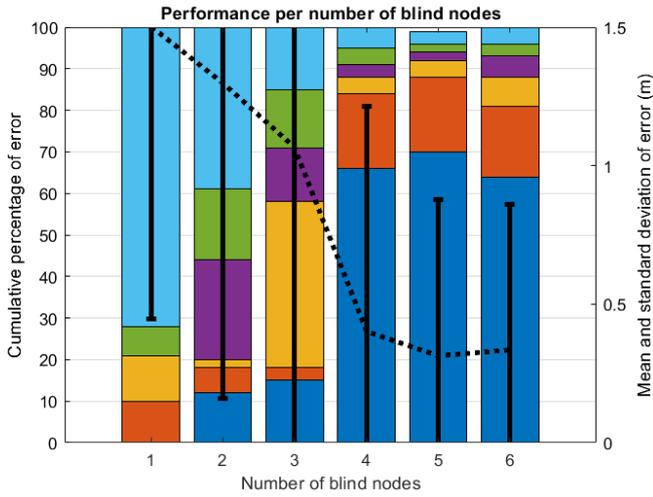
To → From ↓	0	1	2	3	4	5	6	7	8	9
0	-	Succes rate +2.0 % $\mu$ Measured -0.1 m $\sigma$ Measured -0.5 m	Succes rate -1.0 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate -0.8 % $\mu$ Measured -0.1 m $\sigma$ Measured -0.0 m	Succes rate -1.2 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.1 m	Succes rate +1.0 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.1 m	Succes rate -0.8 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +0.2 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate -0.8 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.1 m	Succes rate -1.4 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m
1	Succes rate -1.0 % $\mu$ Measured -0.3 m $\sigma$ Measured -2.9 m	-	Succes rate -1.2 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -1.2 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -2.2 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate +2.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -2.0 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +0.8 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.2 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -1.2 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m
2	Succes rate -2.0 % $\mu$ Measured +0.3 m $\sigma$ Measured +3.0 m	Succes rate +0.0 % $\mu$ Measured - $\mu$ Measured -	-	Succes rate -2.2 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate -2.4 % $\mu$ Measured +0.4 m $\sigma$ Measured +0.1 m	Succes rate +3.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -1.4 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate -0.8 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -2.4 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -2.4 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m
3	Succes rate -0.6 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.2 m	Succes rate -0.2 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -0.6 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	-	Succes rate -1.8 % $\mu$ Measured +0.5 m $\sigma$ Measured +0.2 m	Succes rate +2.2 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -1.6 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.1 m	Succes rate -0.2 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.4 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.4 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m
4	Succes rate -1.4 % $\mu$ Measured +0.5 m $\sigma$ Measured +1.9 m	Succes rate -0.8 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.2 % $\mu$ Measured +0.3 m $\sigma$ Measured +0.1 m	Succes rate -2.8 % $\mu$ Measured +0.4 m $\sigma$ Measured +0.2 m	-	Succes rate +5.2 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.4 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate +0.2 % $\mu$ Measured +0.3 m $\sigma$ Measured +0.1 m	Succes rate -0.8 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.6 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m
5	Succes rate +1.8 % $\mu$ Measured +0.4 m $\sigma$ Measured +1.4 m	Succes rate +2.6 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +2.2 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +2.0 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +1.2 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.0 m	-	Succes rate +0.2 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate +3.4 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate +2.4 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate +0.8 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m
6	Succes rate -3.0 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.6 m	Succes rate -2.2 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -2.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -2.8 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.1 m	Succes rate -2.6 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.0 m	Succes rate -4.0 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	-	Succes rate -1.6 % $\mu$ Measured +0.2 m $\sigma$ Measured +0.0 m	Succes rate -3.0 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -3.2 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m
7	Succes rate -3.2 % $\mu$ Measured -0.1 m $\sigma$ Measured +0.5 m	Succes rate -1.8 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.0 m	Succes rate -2.0 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -2.4 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -2.4 % $\mu$ Measured +0.4 m $\sigma$ Measured +0.1 m	Succes rate -2.2 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -1.4 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.0 m	-	Succes rate -1.2 % $\mu$ Measured +0.3 m $\sigma$ Measured +0.1 m	Succes rate -2.4 % $\mu$ Measured +0.2 m $\sigma$ Measured +0.1 m
8	Succes rate +0.0 % $\mu$ Measured +0.8 m $\sigma$ Measured +4.3 m	Succes rate +2.0 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate +1.6 % $\mu$ Measured -0.0 m $\sigma$ Measured +0.0 m	Succes rate +1.4 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate +22.8 % $\mu$ Measured - $\mu$ Measured -	Succes rate +25.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate +14.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.1 m	Succes rate +10.8 % $\mu$ Measured +0.2 m $\sigma$ Measured +0.1 m	-	Succes rate +6.4 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.1 m
9	Succes rate -4.6 % $\mu$ Measured -0.2 m $\sigma$ Measured -1.0 m	Succes rate -0.6 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -3.6 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -4.2 % $\mu$ Measured +0.0 m $\sigma$ Measured +0.0 m	Succes rate -3.4 % $\mu$ Measured +0.1 m $\sigma$ Measured +0.0 m	Succes rate -2.0 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	Succes rate -3.8 % $\mu$ Measured -0.0 m $\sigma$ Measured -0.0 m	Succes rate -3.8 % $\mu$ Measured +0.2 m $\sigma$ Measured +0.1 m	Succes rate -3.6 % $\mu$ Measured +0.0 m $\sigma$ Measured -0.0 m	-

Table D.2: Quality comparison of the indoors experiment with and without body.

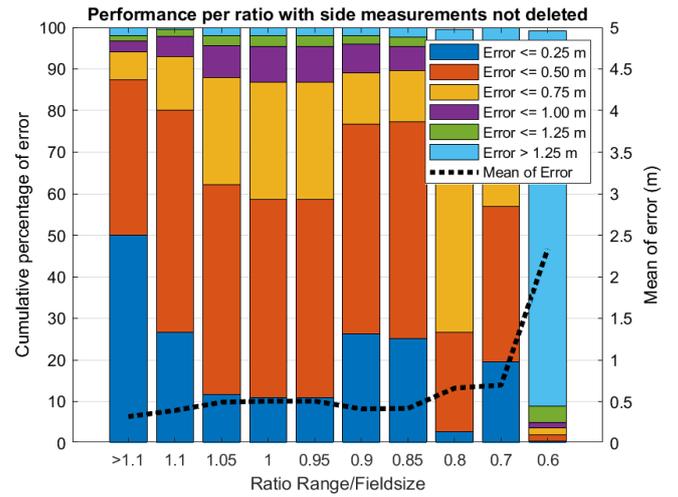
To → From ↓	0	1	2	3	4	5	6
0	-	Succes rate 76.4 % $\mu$ Measured 63.5 m $\sigma$ Measured 0.20 m Real 63.7 m Error 0.3 %	Succes rate 66.0 % $\mu$ Measured 80.4 m $\sigma$ Measured 0.28 m Real 81.0 m Error 0.7 %	Succes rate 0.6 % $\mu$ Measured 118.5 m $\sigma$ Measured 0.32 m Real 118.4 m Error 0.1 %	Succes rate 86.2 % $\mu$ Measured 100.3 m $\sigma$ Measured 0.15 m Real 100.3 m Error 0.0 %	Succes rate 61.0 % $\mu$ Measured 50.0 m $\sigma$ Measured 0.23 m Real 50.2 m Error 0.4 %	Succes rate 86.2 % $\mu$ Measured 59.3 m $\sigma$ Measured 0.10 m Real 59.3 m Error 0.0 %
1	Succes rate 86.6 % $\mu$ Measured 63.9 m $\sigma$ Measured 2.74 m Real 63.7 m Error 0.3 %	-	Succes rate 65.8 % $\mu$ Measured 49.3 m $\sigma$ Measured 0.13 m Real 50.0 m Error 1.4 %	Succes rate 0.4 % $\mu$ Measured 99.6 m $\sigma$ Measured 0.03 m Real 99.8 m Error 0.2 %	Succes rate 85.4 % $\mu$ Measured 118.5 m $\sigma$ Measured 0.09 m Real 118.8 m Error 0.3 %	Succes rate 55.0 % $\mu$ Measured 80.8 m $\sigma$ Measured 0.12 m Real 81.1 m Error 0.4 %	Succes rate 86.0 % $\mu$ Measured 58.5 m $\sigma$ Measured 0.06 m Real 59.3 m Error 1.4 %
2	Succes rate 85.6 % $\mu$ Measured 82.8 m $\sigma$ Measured 7.24 m Real 81.0 m Error 2.2 %	Succes rate 0.6 % $\mu$ Measured 50.2 m $\sigma$ Measured 0.08 m Real 50.0 m Error 0.5 %	-	Succes rate 0.4 % $\mu$ Measured 49.7 m $\sigma$ Measured 0.01 m Real 49.8 m Error 0.1 %	Succes rate 84.4 % $\mu$ Measured 81.2 m $\sigma$ Measured 0.04 m Real 81.2 m Error 0.0 %	Succes rate 43.4 % $\mu$ Measured 64.1 m $\sigma$ Measured 0.08 m Real 63.7 m Error 0.7 %	Succes rate 84.6 % $\mu$ Measured 31.1 m $\sigma$ Measured 0.03 m Real 31.9 m Error 2.2 %
3	Succes rate 83.2 % $\mu$ Measured 123.4 m $\sigma$ Measured 13.91 m Real 118.4 m Error 4.2 %	Succes rate 82.6 % $\mu$ Measured 100.0 m $\sigma$ Measured 0.07 m Real 99.8 m Error 0.2 %	Succes rate 83.4 % $\mu$ Measured 49.5 m $\sigma$ Measured 0.08 m Real 49.8 m Error 0.6 %	-	Succes rate 30.0 % $\mu$ Measured 64.0 m $\sigma$ Measured 0.08 m Real 63.7 m Error 0.4 %	Succes rate 31.4 % $\mu$ Measured 81.3 m $\sigma$ Measured 0.09 m Real 80.7 m Error 0.7 %	Succes rate 0.2 % $\mu$ Measured 58.8 m $\sigma$ Measured 0.00 m Real 59.1 m Error 0.5 %
4	Succes rate 83.4 % $\mu$ Measured 103.2 m $\sigma$ Measured 10.39 m Real 100.3 m Error 2.9 %	Succes rate 82.6 % $\mu$ Measured 118.3 m $\sigma$ Measured 0.09 m Real 118.8 m Error 0.5 %	Succes rate 82.2 % $\mu$ Measured 80.0 m $\sigma$ Measured 0.06 m Real 81.2 m Error 1.5 %	Succes rate 0.2 % $\mu$ Measured 63.3 m $\sigma$ Measured 0.00 m Real 63.7 m Error 0.6 %	-	Succes rate 30.0 % $\mu$ Measured 49.9 m $\sigma$ Measured 0.06 m Real 50.1 m Error 0.3 %	Succes rate 0.2 % $\mu$ Measured 59.3 m $\sigma$ Measured 0.00 m Real 59.5 m Error 0.4 %
5	Succes rate 78.6 % $\mu$ Measured 53.0 m $\sigma$ Measured 10.26 m Real 50.2 m Error 5.6 %	Succes rate 77.4 % $\mu$ Measured 80.7 m $\sigma$ Measured 0.09 m Real 81.1 m Error 0.6 %	Succes rate 73.8 % $\mu$ Measured 62.8 m $\sigma$ Measured 0.09 m Real 63.7 m Error 1.4 %	Succes rate 61.8 % $\mu$ Measured 80.6 m $\sigma$ Measured 0.04 m Real 80.7 m Error 0.1 %	Succes rate 79.0 % $\mu$ Measured 50.1 m $\sigma$ Measured 0.05 m Real 50.1 m Error 0.1 %	-	Succes rate 0.0 % $\mu$ Measured - $\sigma$ Measured - Real 31.9 m Error -
6	Succes rate 86.6 % $\mu$ Measured 63.8 m $\sigma$ Measured 12.46 m Real 59.3 m Error 7.6 %	Succes rate 85.2 % $\mu$ Measured 59.3 m $\sigma$ Measured 0.10 m Real 59.3 m Error 0.0 %	Succes rate 72.2 % $\mu$ Measured 31.4 m $\sigma$ Measured 0.05 m Real 31.9 m Error 1.5 %	Succes rate 0.2 % $\mu$ Measured 59.0 m $\sigma$ Measured 0.00 m Real 59.1 m Error 0.2 %	Succes rate 87.0 % $\mu$ Measured 60.2 m $\sigma$ Measured 0.05 m Real 59.5 m Error 1.1 %	Succes rate 0.6 % $\mu$ Measured 33.0 m $\sigma$ Measured 0.03 m Real 31.9 m Error 3.6 %	-

Table D.3: Quality of the experiment on the large field, device 0-6 are shown here since only those have a known location. Success rates lower than 85 % , a standard deviation of more than 0.50 m and an error of more than 10 % are colored red. Success rates higher than 90 % , a standard deviation of less than 0.03 m and an error of less than 5 % are colored green

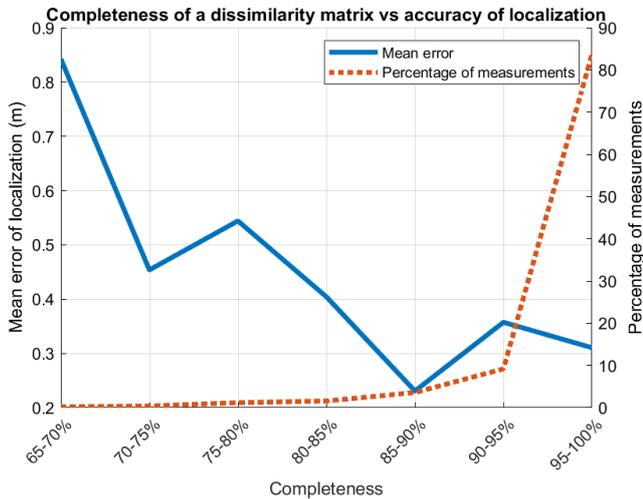
## E More figures for evaluation



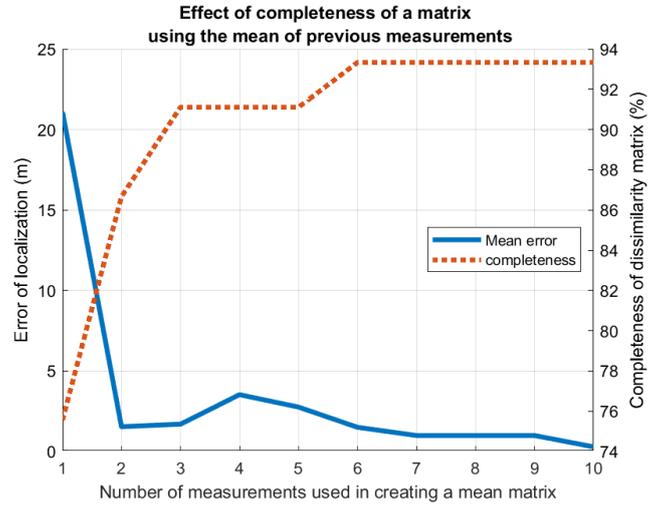
(a) Effect of number of blind nodes on localization performance with an other random initial guess. For the legend, see Figure 7.3



(b) Effect of ratio of  $range/fieldsize$  on localization performance with the side measurements not changed.



(c) Effect of completeness of measurement on localization performance.



(d) Effect of completeness of the dissimilarity matrix using all previous measurements to calculate a mean. Data from the outdoors experiment on a large field is used.

Figure E.1: Figures belonging to section 7

## F Photographs of the experimental setup



(a) A photograph of the indoors experimental setup without a human body.



(b) A photograph of the outdoors experimental setup with a human body.

Figure F.1: Photographs of the experimental setup